International Journal of Process Management and Benchmarking

ISSN online: 1741-816X - ISSN print: 1460-6739
https://www.inderscience.com/ijpmb

Development of assessment framework for process agility in internet of things projects

Madhukar Rajaram Patil, M. Suresh

DOI: 10.1504/IJPMB.2022.10051714

Article History:

Received:	03 July 2022
Accepted:	02 September 2022
Published online:	04 January 2024

Development of assessment framework for process agility in internet of things projects

Madhukar Rajaram Patil and M. Suresh*

Amrita School of Business,
Amrita Vishwa Vidyapeetham,
Coimbatore, 641112, India
Email: madhukarpatil30@gmail.com
Email: m_suresh@cb.amrita.edu
Email: drsureshcontact@gmail.com
*Corresponding author

Abstract: Research and development (R&D) organisations that work on projects connected to the internet of things (IoT) market product designs and build IoT initiatives are called technology IoT organisations. The agility of IoT project organisations is significantly influenced by process agility. Thus, the paper's objective is to assess the IoT projects' process agility level utilising 'multi-grade fuzzy' and 'importance performance analysis (IPA)'. The IoT project's process agility index is calculated using 'multi-grade fuzzy'. It comes to be 8.2, which falls between the range of 8–9 and is designated 'very highly agile'. To increase the process agility of the case IoT project, the IPA is used to discover the weaker attributes. The approach can be applied on an ongoing basis to assist IoT project managers in raising the level of process agility.

Keywords: internet of things; IoT; process agility; IoT projects; agility assessment; multi-grade fuzzy; importance performance analysis; IPA.

Reference to this paper should be made as follows: Patil, M.R. and Suresh, M. (2024) 'Development of assessment framework for process agility in internet of things projects', *Int. J. Process Management and Benchmarking*, Vol. 16, No. 1, pp.36–59.

Biographical notes: Madhukar Rajaram Patil is a research scholar at Amrita School of Business, Amrita Vishwa Vidyapeetham (University), Coimbatore, India and working as Program Manager in Multinational Private Limited Organization, India. He holds a Master of Engineering from University of Pune, Maharashtra State, India and Bachelor of Engineering from Shivaji University, Kolhapur, Maharashtra State, India. He holds 16+ years of industry experience on variety of posts in multiple multinational organisations. His field of study is operations management, with a specific focus on internet of things operations management. His research interests include multi-criteria decision-making, IoT project management, and program management. He is currently working on agility in internet of things domain.

M. Suresh is an Associate Professor at Amrita School of Business, Amrita Vishwa Vidyapeetham, Coimbatore, India. He holds a PhD in Project Management from Indian Institute of Technology, Bombay, India and Master's in Industrial Engineering from PSG College of Technology, Coimbatore, India. His research interests include issues related to lean and agile operations and performance management. He has authored several papers in operations management. He is also a member of International Society on Multiple Criteria Decision-Making.

1 Introduction

Nowadays, agile methodologies are becoming popular program management methodologies in day-to-day execution of programs in software development organisations (Karhatsu et al., 2010). Agile methodologies improve the execution efficiency of the organisation over non-agile organisations. Agile methodologies are widely used in manufacturing industries (Potdar et al., 2017; Sindhwani and Malhotra, 2017, 2018) and supply chains (Galankashi et al., 2019; Matawale et al., 2016). But these agile methodologies are limited to software development project organisation. However, most internet of things (IoT) organisations work on a shorter product development cycle due to changing market environments. Hence agile methodologies need to be fused into the IoT product life cycle. Many important attributes such as customer collaboration, responding to change and incorporating changes during the end-stage of program (Sheffield and Lemétayer, 2013) are key requirements of IoT programs that can be very well handled by using agile methodologies.

IoT projects are kind of projects executed by research and development (R&D) organisations which involve design and development of projects related to IoT markets. Such projects are divided primarily into soft and hard categories (Kisielnicki, 2014). Projects, whose execution is mainly focused on release of design or software code in the form of reports, computer/software models, are classified as soft projects. On the other hand, hard projects involve development of prototypes or actual products. In hard projects, prototypes of actual products are built using released designs or using software codes which demonstrate actual use of the product under design. Final product launch is done post-rigorous validation of prototypes and required testing and certifications on finished products. Compared to soft projects, complexities involved in managing hard projects are much higher. Many methodologies are available separately for end-to-end execution of these programs. With the addition of complexity of IoT products to this, the management of IoT R&D projects is much harder as it involves the management of many other ingredients of the IoT products. Rane and Narvel (2021) gave new perspective of creating IoT programs, using IoT along with block chain for improving agility in Industry 4.0 enablement.

In case of IoT projects, system development includes product hardware design, system software, system firmware, application software, product development and testing. IoT projects compromise lots of system level integration and laboratory testing as well as field testing activities in environment where it will be finally used. IoT projects involve high technology-oriented work and operate in very dynamic work environment. Overall, the adoption of IoT products is increasing very rapidly in market. Hence, there is technological and competitive pressure, which needs IoT organisation to innovate and transform with extremely high pace (Lee and Lee, 2015). The external environment such as digital technology, connectivity and information technology used in IoT products is rapidly changing, due to increased changes in technology. In summary, to cope with increased market requirements of IoT products and rapidly changing market requirements due to changing consumer needs, IoT organisations need to shorten the development cycle of the products. So that IoT organisations can achieve the shortest time to market cycle and succeed against competition.

Apart from regular execution hurdles, IoT project management faces challenges in decision-making, execution methodologies, communication and feedback mechanism.

Unlike software program execution, due care needs to be taken while executing IoT projects as IoT projects need a lot more attention than regular IT/software execution programs.

Some of the key requirements of IoT project execution are:

- early ecosystem partnering to develop infrastructure and standards
- must to have overlap of product execution life cycle with research life cycle
- left shift majority of validation on previous generation products
- close coordination across hardware, software, manufacturing and other support functions such as marketing, regulatory and field teams
- hand in hand execution with involvement of partner ecosystem
- many fold dependencies
- time to market is crucial.
- cost of development needs to be low as profit margins are very low due to competition.

In IoT projects, agile implementation needs to scale up beyond just project team and entire ecosystem needs to implement agile culture.

In IoT project organisation, to bring overall organisational agility, organisation must possess workforce agility, team agility, process agility, engineering/technology agility and business agility. These pillars of agility play a vital role in deriving overall agility of the organisations. Rajan and Dhir (2020) used modified TISM approach for identifying co-relations among factors impacting technology innovations in organisation. Patil and Suresh (2019) have conducted study to identify the importance of workforce agility in technology IoT organisations. Studies show that among many identified factors, top management support and technological infrastructure are top priority factors affecting innovation capabilities of an organisation.

Adaptability, speed of delivery, flexibility, etc. are the key characteristics of agile processes. Agile manifesto gives importance to individuals and interactions over fixed processes and tools. Raschke (2010) identified constructs of process agility. Some of the other important characteristics of process agility are responsiveness, re-configurability, employee adaptability and a process-centric view. Re-configurability and responsiveness focus on the essential features of agility, however employee adaptability and a process-centric view focus on the human aspect of it. Re-configurability characteristic helps to deploy suitable processes which suit the current environment. Responsiveness characteristic gives ability to quickly react according to situation. Third characteristic ‘employee adaptability’ is important to leverage people in changing environment. The last characteristic ‘business process view’ addresses the importance of agility from management perspective, which focus not just on any single domain but on end-to-end entire process. Constantine and Lockwood (2002) explained the benefits of agile practices for achieving goals with changing customer needs in dynamic business environment. The processes can help to define and streamline the roles and tasks. Right processes can be used for prioritising the tasks based on the customer requirements.

Many processes such as XP, Crystal Methods, Scrum, FDM, etc. are being used in the software/IT project executions. All these agile processes are designed in such a way that they can accommodate the changing needs of the customer.

In IoT projects, there are multiple scenarios where customer needs keep changing, hence it must have process agility which can react to these situations and achieve the final accomplishment with minimal or no impact on project schedule, cost and quality. It is necessary for organisation to put efforts on improving internal execution processes and align them to the business growth strategies.

2 Literature review

In software organisations, agile software development methodologies allow end to end software development within self-organising, cross functional, small teams. Agile/scrum teams work closely with customers and deliver working software with quick turnaround. These teams are formed based on the project requirement for short cycles until project completion. In software industry scrum, Lean, XP (extreme programming), Dev Ops, software development, feature driven development, etc. are widely used agile methodologies. Agile software development model promotes an iterative mechanism of software development by means of lifecycle by iterating design-code-test cycle. In a software organisation, agile adaptation motives are decided based on the on-the-end goal. There are two primary categories of agile adaptation in software industry: program management approach and software development approach. Tripp and Armstrong (2018) conducted qualitative analysis of software industry and mapped motives with three different high-level categories: improved quality, efficiency and effectiveness of software. The program management approach focuses more on release planning, iteration planning, burn down, retrospection and velocity. However, software development approach gives more importance to unit testing, automated builds, continuous integration, coding standard, re-factoring and test-driven development. Program management approach improves efficiency and effectiveness; however, software development approach improves quality and effectiveness. As per study conducted by Dybå and Dingsøyr (2008), agile software program management methodologies show major improvements over traditional plan-based software management methods.

In product-based organisations, framework called product lifecycle management (PLM) is used for end-to-end execution of new product development. For successful implementation of PLM, it is necessity to implement PLM in complete ecosystem (Tanpure et al., 2021) In a software organisation, agile methodology implementation is mostly focused on the development of software and people, however in IoT, the concept of agility needs to be much wider, covering people, product hardware, product firmware, product software ecosystem as well as entire environment. Whereas in IoT, focus is not just on processes and people, importance is given to create agility in overall product, organisation and environment.

In IoT products, agile implementation needs to scale up beyond project team and covers entire ecosystem to implement agile culture.

Table 1 Factors influencing process agility in IoT

Sl. no.	*Enabler name*	*Definition/description*	*Reference*
1	Agile planning	Modify the product execution life cycle to suit the specific requirements. The plans need to be flexible enough to accommodate the change request at later point of time.	Conforto and Amaral (2010), Conforto et al. (2014, 2016), Yusuf et al. (1999), Cohn (2005), expert opinion
2	Overlapping development phases	Overlapping of development phases reduces the time required for execution and improves resource planning.	Takeuchi and Nonaka (1986), Roemer et al. (2000), expert opinion
3	Standardisation strategy	Standardise design, which helps in reducing cost and shortening the product development cycle	Albrecht (2018), expert opinion
4	Left shift execution strategy	Shift left is a practice to identify problems and manage potential risk by moving tasks in the product lifecycle to the left as early as possible.	Expert opinion
5	Co-development	Simultaneous exchange of resource and technology during research and development phase of product between other industry partners and development team for common end product.	Perks (2004), expert opinion
6	Co-location	Leverage benefits of quick interactions among team members by co-locating teams.	Paasivaara et al. (2018), Paasivaara and Lassenius (2016), expert opinion
7	Automation	Automation helps in expediting the work and reduces the human dependency in the work. Automation also helps in reducing the defects as human interactions are limited. Automations can be done in design as well as for entire remaining lifecycle of product.	Paasivaara et al. (2018), Van Rossem et al. (2018)
8	Rapid prototyping	Concept of rapid prototype helps design teams to create quick models of final product and do initial testing to validate the first cut use. Rapid prototyping helps design team to try multiple options of final product without wasting time for making final product until it is proved to be useful in first cut tests.	Rabiser et al. (2016, 2018), expert opinion
9	Effective utilisation of time zones	Aligning work split in such a way that it suits different time zones to work within employees own (most productive) time zone.	Expert opinion
10	Reuse strategy	Reuse of standardised items from previous products helps to shorten the product development cycle.	Albrecht (2018), expert opinion
11	Lean methodology	Implementing lean principles will accelerate the product development as it reduces the wasteful efforts. Establish lean culture in the organisation across all the functions.	Fiore (2004), Schulze and Störmer (2012)

Table 1 Factors influencing process agility in IoT (continued)

Sl. no.	*Enabler name*	*Definition/description*	*Reference*
12	Common agile framework	Establishing common agile framework is must within program teams and across organisation. Having common framework will have smooth execution flow. Appropriate agile processes need to be used to implement agile culture effectively.	Paasivaara et al. (2018), Ebert and Paasivaara (2017), Paasivaara and Lassenius (2016), Rigby et al. (2016), Mirza and Datta (2019), Martin and Whitworth (2017), Tirumala et al. (2016), Cooper (2016, 2019), Cooper and Sommer (2016)
13	Customised methodology	Modify the product execution life cycle to suit the specific requirements. The plans need to be flexible enough to accommodate the change request at later point of time. Do not use standardised processes for design, testing as well as development of products. Remove all the idle/non-value adding process steps and personalise them to each product specific.	Conforto and Amaral (2010), Yusuf et al. (1999), expert opinion
14	Customer collaboration	Simultaneous exchange of resource and technology during research and development phase of product between customer and development team. Identify lead customer and develop products simultaneously with them.	Hoda (2011), Hoda et al. (2011a, 2011b), Hoda and Murugesan (2016), Hoda et al. (2010), Perks (2004), Shahin et al. (2017), expert opinion
15	Parallel validation	Conducting parallel validation instead of existing waterfall validation methodology helps to reduce validation throughput time.	Expert opinion
16	Retrospective learning strategy	Overlap development phases in such a way that all the issues/learnings can be well incorporated before final launch	Gelaro et al. (2017), expert opinion
17	Performance against schedule	Monitor the execution program and benchmark and report out the performance against schedule. Reducing total throughput time helps to improve schedule.	Expert opinion
18	Prototype testing	Concept of rapid prototype helps design teams to create quick model of final product and do initial testing to validate the first cut use. Early data from prototype testing helps design team to try multiple options of final product without wasting time for making final product until it is proved to be useful in first cut tests.	Rabiser et al. (2016, 2018), expert opinion
19	Time-cost-quality trade-off	Trade-off decisions to achieve good quality without impact in schedule and cost	Kostami and Rajagopalan (2013)
20	Review and feedback implementation	Plan to introduce maximum reviews before release and expedite validation to expedite to get the required feedbacks. Implement those feedbacks by making incremental changes.	Figueiredo Filho et al. (2021), expert opinion
21	Waste elimination	Eliminate non-value-added legacy features from products to avoid wastage of resources. Moreover remove/reduce non-value added and duplicate tasks from the process to improve schedule performance.	Fiore (2004), Schulze and Störmer (2012), Van Rossem et al. (2019)

2.1 Agile planning

Planning and estimation are important stages of any program (Cohn, 2005). It is not always possible to do complete planning. Planning is iterative and incremental process. Agile planning helps in reducing risk and uncertainty. Better planning supports decision-making process and establishes trust among stakeholders. Planning conveys overall view of the entire project to all the stakeholders.

Plans which are prepared in such way that it can be changed over a course of project execution are termed as agile plans. Every time when change request comes, the impacts are assessed considering whole picture and then decisions are made. Using hypothesis testing, customer needs are proactively collected and ultimately a business plan is developed to meet the expressed needs. Agile planning works on the basis of 'plan-do-adapt' concept. Adopting agile planning helps project teams in encouraging the change requests.

Developing strategic plans for development of new products is key for success of the organisation (Yusuf et al., 1999). Without proper plans, team cannot do monitoring and controlling of the progress of project, which can lead to missing the product launch targets. To achieve the required time to market for launch of products, enterprise-wise integrated project plans need to be developed.

2.2 Overlapping development phases

The product development lifecycle involves multiple phases; every product pass through each of these phases from specifications through customer release (Takeuchi and Nonaka, 1986). In case of sequential approach of product development, transfer of data from one phase to other happens only after tasks of current phase are completed and verified as per phase quality criteria. However, in case of agile product development methodologies, there is an overlap of phases during the execution. Shorter product design lifecycles can be achieved by overlapping the design and development phases (Roemer et al., 2000) This overlapping of development phases reduces the time required for execution and improves resource planning.

Right trade-off strategy needs to be applied between product development time and cost during overlapping of development phases.

2.3 Standardisation Strategy

Traditional product development relies on the creation of new documents. Product requirements, architectural definition, ingredient requirements, and other design and development documents are often captured separately and only loosely related. However, most of the times, there is a possibility of standardisation of these documents from the past records. The records can be kept centrally in library and used as they are or for reference. This standardisation has been proven to be a significant contributor to cost reduction and velocity in project execution.

Standardisation of parts/features and reusing them in recurring products increase organisation efficiencies (Albrecht, 2018).

2.4 Left shift execution strategy

Traditionally, project resource alignment occurs after the complete approvals and execution commits. Shift left is a practice to identify problems and manage potential risk by moving tasks in the product lifecycle to the left as early as possible. Most of the new programs will reuse the basic functionality features from their previous generation programs. Hence target should be taken to complete the requirements documentation matured to fully scoped requirements, reviewed by key program stakeholders with upstream and downstream traceability during early stage of the planning process. Having frozen requirements and clear-cut strategy for execution will allow team to assign the resources and start the execution. Early completion of requirements documentation can enable many execution stages in parallel without need to follow them in sequence.

2.5 Co-development

Simultaneous exchange of resource and technology during R&D phase of the product between other industry partners and the development team is required for common end product. Co-development is basically next level of customer – collaboration, wherein customer as well as sub-vendors works together for launching new technology product to market such as launching 5G smartphones. Co-development will always be done with multi-party contracts, wherein customer as well as supplier/vendor is abided to work only within themselves and protect each other's intellectual properties. Co-development of product needs intensive collaboration among all the partners who are involved in the development of the product (Perks, 2004). Even for some technology innovations, co-development happens closely between competing large organisations. The effectiveness of co-development depends on the knowledge and resource sharing models.

2.6 Collocation

Today most of the organisations operate globally and there are quite good number of resources who work from different physical locations across the world. This is mainly because competencies of each domain may not be located at the same place (Paasivaara et al., 2018). However, this kind of multi-location project execution creates delay in execution. This impact of delay is quite high during early stage of the program. However, this delay can be minimised if key resources can ideally work in same physical location like a Dungeon Room until they define complete execution strategy. Collocation enables quick interactions among team members, reducing time to converge on topics. Physical co-location needs stakeholders to travel from different location to one single place, increasing cost of execution.

2.7 Automation

In the early phase of program execution during the prototype design and development, most of the processes are manual. However, these manual processes are major hurdles for continuous deployment (Paasivaara et al., 2018). Hence wherever possible, there is a need to do automation in these processes. With the advancements in telecom and infrastructure (Van Rossem et al., 2018), it has become much easier to make automations.

Automation helps in expediting the work and reducing the human dependency at the workplace. Repetitive work can be converted from human touch to machine touch. Automation also helps in reducing the defects as human interactions are limited. There are many advances in technologies which help organisation to automate the tasks. Automations can be done in design as well as for entire remaining lifecycle of product. Infrastructure plays a vital role in enabling automation (Van Rossem et al., 2018). With the evolution of technology in electronics, computer, information technology and internet technologies, many tasks can be automated and remotely monitored with minimal human touch. With the help of right infrastructure, continuous development of products with best quality and shorter time to market can be easy (Paasivaara et al., 2018). Automation can be enabled in design, manufacturing, testing in many other intermediate functions too.

Van Rossem et al. (2019) have explained methodology of virtual compute and network resource allocation model which increases the flexibility of resource allocation and in turn increases the execution speed.

2.8 Rapid prototyping

Designs do not evolve into fully functional product just in one iteration. However due to long manufacturing lead times, these iterations take longer times. During design phase, to evaluate the design options, quick manufacturing or working prototypes can be done to perform the required evaluation.

Design thinking (Abdulmanova et al., 2019; Tschimmel, 2012; Carlgren et al., 2016; Henriksen et al., 2017) through creation of rapid prototypes helps to solve unstructured problems and incubate innovations (Glen et al., 2015). Prototyping of any idea generates tangible visualisation of concept of rapid prototype helps design teams to create quick model of final product and do initial testing to validate the first cut use. Clones are commonly used to create derivative products (Rabiser et al., 2016, 2018). Clones can be created at individual feature level or at full product level. Prototypes and clones are used widely in reverse engineering and in re-factoring.

Quick prototypes are created to make variants of main product to start the early validation (Rabiser et al., 2016, 2018).

2.9 Effective utilisation of time zones

With most organisations going global, the projects are executed with a mix of team members across globe. Ideally each project team executing the program should have all the required functions co-located at same place, however that is not always possible due to specific skill set requirement which is not possible to co-locate or other difficulties such as need to co-locate few functions close to customer place. In such scenarios, it becomes very challenging to work across all the time zones. In such cases, most of the cross-GEO meetings and tasks need to be aligned in GEO friendly time. However, the attendance from non-friendly time zone GEO can be limited to only leads, so that not all the team members have to face the time zone challenge. In certain type of engagements/work, wherein work needs to happen 24/7, teams working of different time zones help to improve the productivity and execution speed. In such engagements, proper overlaps are needed to transfer updates and data on daily basis. Aligning work split in such a way that it suits different time zones to work within employees own (most productive) time zone increases the agility.

2.10 Reuse strategy

Traditional product development relies on the creation of new documents. Product requirements, architectural definition, ingredient requirements, and other design and development documents are often captured separately and only loosely related. However, most of the times, there is a possibility of reusing these documents from the past records. The records can be kept centrally in library and used as is or for reference from the central library. This reuse has been proven to be a significant contributor to velocity in project execution.

Reuse of standardised items from previous products will shorten the product development cycle. Reuse can drive efficiencies and ultimately reduce time to market. Reuse common requirements from existing programs to new programs. Efficiency of reuse can be analysed by using term reuse enable velocity. Standardisation of parts/features and reusing them in recurring products increase organisation efficiencies (Albrecht, 2018).

2.11 Lean methodology

Most of the times, projects are developed with the baseline from existing projects, hence they reuse the existing design methodologies. However, there could be some stages in existing design methodologies which become redundant and may not require for designing new product. These redundant stages basically generate waste during execution. By using lean methodology, efforts should be taken to remove/reduce non-value added and duplicate tasks from the processes to improve schedule performance. Support functions in the organisation should plan their tasks appropriately. Lean methodology, in turn, works on the waste elimination and cycle time reduction through proper flow and process-based approach (Fiore, 2004). Implementing lean principles will accelerate the product development as it reduces the wasteful efforts.

Lean start-ups (Reis, 2011) are concept used in software industry for organisations which are emerging as start-up organisations with inbuilt lean culture (Müller and Thoring, 2012). For successful implementations of lean culture, focus needs to be put on employee training and coaching and constructive failure treatment (Schulze and Störmer, 2012)

2.12 Common agile framework

Agile methodologies are becoming popular in large organisations (Dikert et al., 2016). Large-scale agile implementation requires agility to be implemented in all the teams involved in development. All the development teams need to synchronise and interface with each other for scaling the agile at large organisational level (Paasivaara et al., 2018). Establishing common agile framework is a must within program teams and across the organisation. Standard procedures, indicators and other frameworks need to be developed and followed across the organisation. Having common framework will have smooth execution flow across the organisation.

2.13 Customised methodology

The common agile framework helps in smooth execution of the processes, however sometimes it generates waste in the system as it may have some redundant stages which are unwanted. Simple alterations can be done to the existing methodology to suit the new requirements while keeping in mind the overall common agile framework. Iterative agile methodologies have been identified to be beneficial over current best project management practices (Conforto and Amaral, 2010; Conforto et al., 2016). The product execution life cycle is modified to suit the specific requirements. The plans need to be flexible enough to accommodate the change request at later point of time.

2.14 Customer collaboration

Every product has been developed considering one or more intended use. These requirements are then converted to engineering specifications to develop products. The products developed by organisation should suffice the customer needs. Customer needs are dynamic and change over time. Theoretically, these customers needs follow Kano model, which means over a period of time, the esteem needs become basic needs.

Customer collaboration during execution is a key parameter in success of agile project (Hoda et al., 2011a). Organisations collaborate with their industry partners for long-term new product developments (Perks, 2004). Even resources can be exchanged between organisations to increase the collaboration.

Hypothesis testing and agile methodology are combined to create the project planning. Using hypothesis testing, the team can detach from their own beliefs and listen to customers, learning about their needs and ultimately developing a business plan to meet the expressed needs. During the research, it was observed that teams agreed on the point that it is ‘a business imperative’ that they listen to the customers, and using this ‘scientific method’ of hypothesis testing helps them to detach from their own beliefs and listen to the customers. Quick feedback is required in agile projects. Customer collaboration helps in achieving continuous software engineering which includes continuous integration, continuous delivery and continuous deployment (Shahin et al., 2017). The effectiveness of collaboration depends on the knowledge and resource sharing models.

2.15 Parallel validation

Validation is an extremely important phase in the overall product lifecycle. Validation is conducted in two phases: pre-product phase and post-product phase on actual parts. Pre-product validation can be done in two parts: simulation and prototype testing/emulation. Simulation is the method of validation which can be performed on designs which are in early design stages. Simulation is performed using high performance computers and specific software programs. Emulation or prototype testing can be done on more developed designs before the final design release for manufacturing. Compared to simulation, emulation/prototype testing takes more time and is limited by number of prototypes or emulation platforms. As the real products are going to be available for testing, post-manufacturing validation can be done on actual products. During the product development cycle, validation on real products can be limited by availability of number of working test stations and the features enabled on the new product.

Simulations can be done on multiple design options to choose right design; however, it will be limited by number of people working on it and availability of high-performance computing (HPC). Efforts are needed to add more people and additional engineering computing so that validation can be run in tandem to reduce the overall simulation time for the project. Similarly, emulation is limited by availability of emulation setups, hence care needs to be taken to add required number of setups to conduct emulation in tandem. Initially, validation on actual manufactured parts is started in labs using lab test setups. Post successful lab testing and necessary regulatory approvals, parts are sent to customers for next level testing and then field testing can be conducted. Not all these tests need to happen in waterfall way, most of them can start in parallel, so that complete validation cycle can be finished faster. Conducting parallel validation instead of existing waterfall validation methodology helps to reduce validation throughput time.

2.16 Retrospective learning strategy

Gelaro et al. (2017) explained the best use of retrospective study in research and applications. Re-analysis can be done on the past data which is collected over prescribed span of time. Data assimilation can be done to understand the variation of actual data over forecast. The devices used for this retrospective study data collection can be early prototypes of new design or the previous generation similar products. If the previous generation similar products are not available, product lifecycles are designed in such a way that one or more iterations are included as a part of product lifecycle. Though all the design verifications are done using simulation tools prior to release of design for manufacturing, in this stage, real time testing are conducted in labs to validate the intended uses. All the issues are identified and tracked to get fixed in \next design releases. Though retrospective learning strategy increases design cycle time, it improves quality of product as issues are identified and fixed prior to market releases.

2.17 Performance against Schedule

Project schedules are prepared at the starting of the program. Basically, schedule provides information about the tasks, its duration, start date and finish date. In the schedule, tasks are interconnected to each other with constraints and their predecessors and successors. Any change in duration or start of predecessor impacts the start and completion of successor task. If there is any push out of one of the tasks, all the tasks dependent on this task get delayed. Schedule also helps us to identify the critical path for the program. Every project can have one or more critical paths. Delay in any of the tasks of critical path impacts the overall duration of project. Critical paths are not fixed in the program schedule, they keep on changing as per the current situation. All the tasks are executed according to the schedule. Monitoring and controlling of tasks are done using the project schedule. Schedules are used to represent the overall progress of project. Comparison of the projects is done by benchmarking of schedules of other reference projects. Schedules are used to report out the performance against schedule. There are many ways to expedite or fast forward the schedule, schedule crushing is one of that method. Reducing total throughput time will help to improve schedule.

2.18 Prototype testing

In some cases, products are completely designed from the scratch. In such cases there is not any reference from previous generation products. Hence, it is a must to create some quick prototypes and test them for intended use to identify design issues prior to its market launch. Concept of rapid prototype helps design teams to create quick models of final product and do initial testing to validate the first cut use cases. Early data from prototype testing helps design teams to try multiple options to finalise product without wasting time for making final product until it is proved to be useful in first cut tests. Clones from previous generation products are used either at product level or feature level to do required testing (Rabiser et al., 2016). The testing data collected from prototype testing can be used to validate the proposed concept or to collect the data to do retrospective analysis.

2.19 Time-cost-quality trade-off

The trade-off decision methodology provides strategy to conceptualise the improvement methodologies (Da Silveira and Slack, 2001). The trade-off can be done based on importance and sensitivity of factors influencing the decision. In product development, organisation trade-off decisions are made to achieve good product quality without impact in schedule and cost (Zhang and Xing, 2010). There can be multiple scenarios where organisation needs to take decisions based on trade-off (Kostami and Rajagopalan, 2014). The scenarios for trade-off analysis arise when the organisation wants to expedite the product development within given cost without impacting the quality of the product (Meier et al., 2016). Multiple customised tools are available in market to conduct the trade-off analysis. For implementation of trade-off decisions, many times automations or innovations can be implemented to make sure that the change of one parameter does not negatively impacts other parameters.

2.20 Review and feedback implementation

Early engagement of key stakeholders and proactive communications with stakeholders improve probability of identification of opportunities and risks and create most effective contingency strategies for better program success probability (Figueiredo Filho et al., 2021). Peer reviews and formal release reviews are the most important stages of every design cycle. Reviews provide platform for discussing cross questions and inputs from reviewers. Better to add more and more reviews to refine the product before its release in the market. It is thus recommended to design a product lifecycle to introduce maximum reviews before its release and expedite validation to get the required feedback. Post-review feedback needs to be properly tracked until they are dispositioned. The disposition of review findings/feedbacks can be either to implement them or to reject them with appropriate justification. Many organisations use different software to track the review findings/feedback to avoid any misses and for future references.

2.21 Waste elimination

In new product design and development, the requirements would not be fully finalised before start of execution. In high volume manufacturing, requirements come in terms of final released drawings, however in case of new product development, requirements come in terms of specifications, which keep evolving over period of time (Mascitelli, 2007). Hence, in new product development, more interactions among different stakeholders are needed during the execution, which sometimes lead to lot of wastages. While we inherently recognise that duplication is waste, a quick look-around reveals duplication occurring in many processes. The duplication can occur between software development teams and validation engineering teams. Since these two systems of record do not communicate efficiently and effectively, validation engineers often duplicate the work that is already there in designer's system of records. Efforts should be taken to centralise the design documentation and reuse them as maximum as possible.

3 Research methodology

3.1 Multi-grade fuzzy

The multi-grade fuzzy technique was used to evaluate the success of procurement management as well as agile, lean, and efficiency in the industrial and service sectors (Vinodh and Aravindraj, 2015; Sridharan and Suresh, 2016; Ganesh and Suresh, 2016; Vinodh and Chintha, 2011; Vinodh, 2011; Vimal et al., 2015; Almutairi et al., 2019, Patil and Suresh, 2021).

This research was conducted using multi-grade fuzzy to evaluate the IoT projects' process agility. The current study begins with the literature review on process agility in IoT projects and assessment of process agility level in the IoT projects. In Table 2, a new conceptual model is presented with 4 enablers, 8 criteria, and 21 attributes to evaluate the process agility index.

Table 2 Conceptual model of process agility of IoT projects

Enablers	*Criteria*	*Attributes*
Execution strategy (P1)	Program planning (P11)	Agile planning (P111)
		Overlapping development phases (P112)
		Standardisation strategy (P113)
		Left shift execution strategy (P114)
	Working strategy (P12)	Co-development (P121)
		Co-location (P122)
Advance process technology (P2)	Technology measures (P21)	Automation (P211)
		Rapid prototyping (P212)
	Operational measures (P22)	Effective utilisation of time zones (P221)
		Reuse strategy (P222)
		Lean methodology (P223)

Table 2 Conceptual model of process agility of IoT projects (continued)

Enablers	*Criteria*	*Attributes*
Methodology (P3)	Execution framework (P31)	Common agile framework (P311)
		Customised methodology (P312)
	Feedback strategy (P32)	Customer collaboration (P321)
		Parallel validation (P322)
		Retrospective learning strategy (P323)
Monitoring and controls (P4)	Evaluation methodology (P41)	Performance against schedule (P411)
		Prototype testing (P412)
	Corrective actions (P42)	Time-cost-quality trade-off (P421)
		Review and feedback implementation (P422)
		Waste elimination (P423)

4 Case study

4.1 Case project

The case IoT project organisation is a multinational company that has an office located in India. The case organisation manages numerous IoT projects in different countries. For projects to be completed on time and delivered effectively, process agility is essential. The level of process agility will be evaluated using a case study IoT project.

The case IoT project's process agility assessment index is denoted by the letter *P*. It is the product of the overall assessment level based on ratings of each driver (*R*) and the overall weights (*W*) given by the experts. The equation for process agility index is (Suresh et al., 2020):

$$P = W \times R$$

The assessment has been divided into ten grades and the entire process agility index involves fuzzy determination. $P = \{10, 9, 8, 7, 6, 5, 4, 3, 2, 1\}$. 9–10 represents 'extremely agile', 8–9 represents 'very highly agile', 7–8 represents 'highly agile', 6–7 represents 'agile', 5–6 represents 'fairly agile', 4–5 represents 'low agile', 3–4 represents 'very low agile', 2–3 represents 'moderately not agile', 1–2 represents 'not agile', and less than 1 represents 'extremely not agile'. A questionnaire is used with a ten-point Likert scale to collect data (for ratings and weightage) from experts in IoT projects. The weightage has been given by five experts from different IoT projects organisations in India. The performance ratings are collected from five experts in the case-IoT project and it is captured in Table 3.

4.1.1 Primary assessment calculation

The primary calculation done for the 'program planning (P11)' is given below:

- Weights concerning to 'program planning' criterion is W_{11} = [0.251, 0.227, 0.245, 0.276].

Table 3 Weights and performance rating from experts

Pi	*Pij*	*Pijk*	*Experts rating*					*Normalised weightage*		
			E1	*E2*	*E3*	*E4*	*E5*	*Wij*	*Wj*	*W*
P1	P11	P111	9	8	8	7	9	0.251534	0.5111	0.2721
		P112	8	7	8	7	7	0.226994		
		P113	10	8	8	9	8	0.245399		
		P114	8	10	9	9	10	0.276074		
	P12	P121	9	9	8	9	9	0.547619	0.4888	
		P122	7	8	7	7	7	0.452381		
P2	P21	P211	7	9	9	8	8	0.47619	0.4943	0.2603
		P212	9	9	10	10	8	0.52381		
	P22	P221	8	8	7	8	7	0.328	0.5056	
		P222	10	8	8	9	8	0.32		
		P223	8	8	9	9	8	0.352		
P3	P31	P311	8	7	7	8	8	0.4875	0.5113	0.2603
		P312	9	8	8	7	9	0.5125		
	P32	P321	9	10	9	8	9	0.368	0.4886	
		P322	8	7	8	7	9	0.336		
		P323	8	8	7	7	8	0.296		
P4	P41	P411	7	7	7	8	7	0.435897	0.5	0.2071
		P412	9	9	10	10	8	0.564103		
	P42	P421	7	7	8	7	8	0.283333	0.5	
		P422	8	7	8	7	9	0.35		
		P423	8	8	9	9	8	0.366667		

Assessment for the performance of 'program planning' criterion is given below as:

$$R_{11} = \begin{bmatrix} 9 & 8 & 8 & 7 & 9 \\ 8 & 7 & 8 & 7 & 7 \\ 10 & 8 & 8 & 9 & 8 \\ 8 & 10 & 9 & 9 & 10 \end{bmatrix}$$

Index concerning of 'program planning' criterion is given by

$$P_{11} = W_{11} \times R_{11}$$

$$P_{11} = [8.74, 8.325, 8.276, 8.043, 8.576]$$

Utilising the above principle, the index concerning for the following criteria in process agility assessment are obtained and given below.

$$P_{12} = [8.095, 8.547, 7.547, 8.095, 8.095]$$

$$P_{21} = [8.047, 9, 9.524, 9.047, 8]$$

$$P_{22} = [8.64, 8, 8.024, 8.672, 7.672]$$

$$P_{31} = [8.512, 7.51, 7.51, 7.487, 8.512]$$

$$P_{32} = [8.368, 8.4, 8.072, 7.37, 8.7]$$

$$P_{41} = [8.13, 8.13, 8.69, 9.13, 7.564]$$

$$P_{42} = [7.72, 7.37, 8.37, 7.73, 8.35]$$

4.1.2 Secondary assessment calculation

The calculation concerning to enabler of 'execution strategy (P1)' is given below:

- Weights concerning to 'execution strategy' enabler is given as $W_1 = [0.511, 0.489]$.

Assessment for the performance of 'execution strategy' enabler is given below:

$$R_1 = \begin{bmatrix} 8.74 & 8.325 & 8.276 & 8.043 & 8.576 \\ 8.095 & 8.547 & 7.547 & 8.095 & 8.095 \end{bmatrix}$$

Index concerning of 'execution strategy' enabler is given by:

$$P_1 = W_1 \times R_1$$

$$P_1 = [8.425, 8.434, 7.92, 8.34]$$

Utilising the above principle, the index concerning for the following enabler in process agility assessment are obtained and given below.

$$P_2 = [8.347, 8.494, 8.857, 7.834]$$

$$P_3 = [8.44, 7.95, 7.785, 8.6]$$

$$P_4 = [7.92, 7.75, 8.53, 8.43, 7.96]$$

4.1.3 Tertiary assessment calculation

The assessment value of the process agility of case IoT project has been calculated as follows:

Complete weight $W = [0.272, 0.26, 0.26, 0.207]$

$$\text{Complete assessment vector } R = \begin{bmatrix} 8.425 & 8.43 & 7.92 & 8.07 & 8.34 \\ 8.347 & 8.49 & 8.76 & 8.86 & 7.83 \\ 8.44 & 7.95 & 7.78 & 7.43 & 8.6 \\ 7.92 & 7.75 & 8.53 & 8.43 & 7.96 \end{bmatrix}$$

Process agility index $P = W \times R$

$$P = [8.3, 8.18, 8.23, 8.18, 8.19]$$

The final process agility index is the average of $P = 8.2 \in$ (8 to 9). Therefore, it is labelled as 'very highly agile'.

4.2 Importance performance analysis

Importance performance analysis (IPA) is widely used in manufacturing and service industries for classifying the attributes based on the importance and performance (Martilla and James, 1977; Menon and Suresh, 2020; Anil and Suresh, 2020). In IPA, the *x*-axis is the performance of the attributes and the *y*-axis is the importance. The mean of the *x*-axis is 8.17 and the mean of *y*-axis is 8.18 as shown as a perpendicular line in Figure 1.

Figure 1 IPA analysis for process agility for case IoT project

Importance ↑												
9.2	Quadrant I							Quadrant II		P121	P321	
9												P114
8.8								P223, P423				P212, P412
8.6												
8.4				P322, P422								
8.2			P221			P111, P312						
8						P211			P113, P222			
7.8			P311									
7.6	P122											
7.4		P112	P323									
7.2												
7												
6.8	P411	P421	Quadrant IV								Quadrant III	
	7.2	7.4	7.6	7.8	8		8.2	8.4	8.6	8.8	9	9.2
	Performance Rating →											

- *Quadrant I (Concentrate here):* The IoT case project manager must pay close attention to the characteristics in the quadrant if they want to enhance their effectiveness. The attributes are agile planning, effective utilisation of time zones, customised methodology, parallel validation, and review and feedback implementation.
- *Quadrant II (Keep up the good work):* The attributes in this quadrant are needed to be maintained. The attributes are left shift execution strategy, co-development, rapid prototyping, lean methodology, customer collaboration, prototype testing, and waste elimination.
- *Quadrant III (Possible overkill):* The attributes in this quadrant have low importance but high performance. The attributes are standardisation strategy and reuse strategy.

- *Quadrant IV (Low priority):* The attributes in this quadrant have low importance and low performance. The attributes are overlapping development phases, co-location, automation, common agile framework, retrospective learning strategy, performance against schedule, and time-cost-quality trade-off.

5 Results and discussion

The case IoT project's process agility index calculated from this model is 8.2, indicating that the IoT project's process execution is 'very high agile'. Further research utilising the 'IPA' approach revealed that in order to achieve significantly better agility in the IoT process within the firm, several attributes require rapid attention from top management.

Organisations must put plans in place to increase process efficiency by using measures including agile planning, efficient use of time zones, tailored methodology, parallel validation, review, and feedback.

IPA methodology is used in the current case study to identify the weaker attributes by analysing performance along with the importance. The IoT firms need to concentrate on their advantages while paying close attention to the crucial areas for development. Table 4 provides recommendations for strengthening the case IoT project's weaker attributes.

Table 4 Suggestions for weaker attributes

Weaker attributes	*Suggestions for improvement*
Agile planning	• Focus on preparing plans which can be modified easily upon needed • Continuously monitor the progress and tweak the plans • In addition to short-term execution plans, prepare long-term future looking plans • Create integrated plans to view the full picture of market launch
Effective utilisation of time zones	• Identify the right teams to get benefit of different time zones • Automate to minimise the daily work transition time
Customised methodology	• Focus on minimising waste due to usage of standard/universal methodologies and tools • Simplify the methodology to make it more specific to requirement and minimise cycle time
Parallel validation	• Plan the validation resources, both human and non-human, appropriately to carry the fast-track validation • Identify the ways to incorporate parallel validation as maximum as possible • Focus on utilising the benefits of customer/partner lab infrastructure to expedite the testing
Review and feedback implementation	• Plan internal regulatory body for formal reviews • Cross the project team boundaries and identify the reviewers for peer reviews • Track the issues identified in all the internal as well as external reviews and audits until they are dispositioned appropriately

6 Practical implication

Organisations need to focus on overall planning of the programs. They need to put efforts to create more comprehensive plans for immediate programs and tentative future looking long-term plans for upcoming programs to make sure the overall organisation level prioritisation decisions can be taken without any delays. The program plans need to be flexible enough so that they can be modified appropriately based on the work progress and change requests, if any. Project team needs to integrate all the individual feature/component plans at one place, so that management can get overall picture of the progress of the project and intervene to make necessary execution changes to make sure the project is on track.

While selecting project execution team, attention should be given to split the work appropriately so that effective utilisation of time zones can be used to get maximum benefit and reduce the overall schedule of the project. The transition of work from one time zone to other needs to be made as quickly as possible, moreover, if possible, needs to be automated to avoid any loss in productivity due to transition.

Design methodologies evolve over a period of time, thus it should be made sure that efforts are taken to put the right methodology in place, before start of full execution. If needed, pilot runs should be done to resolve any minor issues before full swing execution. With the right customised methodology, waste will be reduced as well as the execution will be faster.

Thorough validation of product prior to launch gives high confidence in product functionality for the intended use. It must be ensured that every feature/use case is validated thoroughly as per standards. However, validation is always a long pole in any design cycle. Thus, appropriate planning is needed to include parallel validation to make sure the validation is executed in fast mode. However, this needs more resources, hence it must be planned efficiently to utilise the resources, manpower and infrastructure to carry the fast-track validation. Wherever possible, collaborate with external labs and customers to conduct the validation in parallel. Conduct detailed reviews and audits to get feedback about the work. Project plans should be made to accommodate maximum possible reviews and audits. Organisations need to plan internal regulatory body for formal reviews. It is always better to call reviewers/experts from other business groups, as quality of peer reviews is better if they are done without any bias.

7 Conclusions

The process agility is crucial in determining the agility of IoT projects. Utilising literature reviews and expert comments, the enablers, criteria, and attributes of the process agility are determined. The IoT project’s present level might be determined by measuring the process agility, which would also help to increase the project’s overall agility. The development of the process agility assessment framework uses the ‘multi-grade fuzzy approach’. The case IoT project’s process agility index is 8.2, which indicates that team executing IoT project is ‘very high agile’. The IPA is used to categorise the attributes and identify the project’s case’s weaker features. For achieving better process agility, the organisation has to do due diligence to make good quality overall program plans, implement right operational and technological measures, and make sure the right execution framework has been set along with proper feedback strategy. Organisation has to make sure the focus is given to identify the issues through validation and reviews prior to launch of programs. Every feedback is tracked properly until it has been dispositioned appropriately (accepted/implemented or rejected with justification). Future systems that support decision-making in process agility evaluation in IoT projects may be created using fuzzy logic and artificial intelligence techniques.

References

Abdulmanova, T., Davlenova, A., Alkuatova, A., Pastuszak, Z. and Turkyilmaz, A. (2019) 'Improvement of design thinking process using Kaizen tools. Thriving on future education, industry, business and society', *Proceedings of the MakeLearn and TIIM International Conference 2019*, ToKnowPress, pp.337–337.

Albrecht, D.J. (2018) 'Product standardization and product design modularity: a case study of motorola and the mobile handset market', *The Changing Landscape of Global Businesses: Principles and Practices*, pp.167–182.

Almutairi, A.M., Salonitis, K. and Al-Ashaab, A. (2019) 'Assessing the leanness of a supply chain using multi-grade fuzzy logic: a health-care case study', *International Journal of Lean Six Sigma*, Vol. 10, No. 1, pp.81–105.

Anil, M. and Suresh, M. (2020) 'Assessment of service agility in power distribution company', *IOP Conference Series: Materials Science and Engineering*, Vol. 954, No. 1, p.012010.

Carlgren, L., Rauth, I. and Elmquist, M. (2016) 'Framing design thinking: the concept in idea and enactment', *Creativity and Innovation Management*, Vol. 25, No. 1, pp.38–57.

Cohn, M. (2005) *Agile Estimating and Planning*, Pearson Education, Upper Saddle River, New Jersey.

Conforto, E.C. and Amaral, D.C. (2010) 'Evaluating an agile method for planning and controlling innovative projects', *Project Management Journal*, Vol. 41, No. 2, pp.73–80.

Conforto, E.C., Amaral, D.C., da Silva, S.L., Di Felippo, A. and Kamikawachi, D.S.L. (2016) 'The agility construct on project management theory', *International Journal of Project Management*, Vol. 34, No. 4, pp.660–674.

Conforto, E.C., Salum, F., Amaral, D.C., Da Silva, S.L. and De Almeida, L.F.M. (2014) 'Can agile project management be adopted by industries other than software development?', *Project Management Journal*, Vol. 45, No. 3, pp.21–34.

Constantine, L.L. and Lockwood, L. (2002) 'Process agility and software usability: toward lightweight usage-centered design', *Information Age*, Vol. 8, No. 8, pp.1–10.

Cooper, R.G. (2016) 'Agile–stage-gate hybrids: the next stage for product development blending agile and stage-gate methods can provide flexibility, speed, and improved communication in new-product development', *Research-Technology Management*, Vol. 59, No. 1, pp.21–29.

Cooper, R.G. (2019) 'The drivers of success in new-product development', *Industrial Marketing Management*, Vol. 76, pp.36–47, https://doi.org/10.1016/j.indmarman.2018.07.005.

Cooper, R.G. and Sommer, A.F. (2016) 'The agile–stage-gate hybrid model: a promising new approach and a new research opportunity', *Journal of Product Innovation Management*, Vol. 33, No. 5, pp.513–526.

Da Silveira, G. and Slack, N. (2001) 'Exploring the trade-off concept', *International Journal of Operations & Production Management*, Vol. 21, No. 7, pp.949–964, https://doi.org/10.1108/01443570110393432.

Dikert, K., Paasivaara, M. and Lassenius, C. (2016) 'Challenges and success factors for large-scale agile transformations: a systematic literature review', *Journal of Systems and Software*, Vol. 119, pp.87–108, https://doi.org/10.1016/j.jss.2016.06.013.

Dybå, T. and Dingsøyr, T. (2008) 'Empirical studies of agile software development: a systematic review', *Information and Software Technology*, Vol. 50, Nos. 9–10, pp.833–859.

Ebert, C. and Paasivaara, M. (2017) 'Scaling agile', *IEEE Software*, Vol. 34, No. 6, pp.98–103.

Figueiredo Filho, L.B.G., Bouzon, M. and de Castro Fettermann, D. (2021) 'An analysis of the effects of stakeholders management on IT project risks using Delphi and design of experiments methods', *Benchmarking: An International Journal*, Vol. 29, No. 3, pp.713–734.

Fiore, C. (2004) *Accelerated Product Development: Combining Lean and Six Sigma for Peak Performance*, Productivity Press, New York.

Galankashi, M.R., Helmi, S.A., Rahim, A.R.A. and Rafiei, F.M. (2019) 'Agility assessment in manufacturing companies', *Benchmarking: An International Journal*, Vol. 26, No. 7, pp.2081–2104, https://doi.org/10.1108/BIJ-10-2018-0328.

Ganesh, J. and Suresh, M. (2016) 'Safety practice level assessment using multigrade fuzzy approach: a case of Indian manufacturing company', in *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, IEEE, pp.1–5.

Gelaro, R., McCarty, W., Suárez, M.J., Todling, R., Molod, A., Takacs, L. and Zhao, B. (2017) 'The modern-era retrospective analysis for research and applications, version 2 (MERRA-2)', *Journal of Climate*, Vol. 30, No. 14, pp.5419–5454.

Glen, R., Suciu, C., Baughn, C.C. and Anson, R. (2015) 'Teaching design thinking in business schools', *The International Journal of Management Education*, Vol. 13, No. 2, pp.182–192.

Henriksen, D., Richardson, C. and Mehta, R. (2017) 'Design thinking: a creative approach to educational problems of practice', *Thinking skills and Creativity*, Vol. 26, pp.140–153, https://doi.org/10.1016/j.tsc.2017.10.001.

Hoda, R. (2011) *Self-organizing Agile Teams: A Grounded Theory*, New Zealand.

Hoda, R. and Murugesan, L.K. (2016) 'Multi-level agile project management challenges: a self-organizing team perspective', *Journal of Systems and Software*, Vol. 117, pp.245–257, https://doi.org/10.1016/j.jss.2016.02.049.

Hoda, R., Noble, J. and Marshall, S. (2010) 'Organizing self-organizing teams', *2010 ACM/IEEE 32nd International Conference on Software Engineering*, IEEE, Vol. 1, pp.285–294.

Hoda, R., Noble, J. and Marshall, S. (2011a) 'The impact of inadequate customer collaboration on self-organizing agile teams', *Information and Software Technology*, Vol. 53, No. 5, pp.521–534.

Hoda, R., Noble, J. and Marshall, S. (2011b) 'Supporting self-organizing agile teams', *International Conference on Agile Software Development*, Springer, Berlin, Heidelberg, pp.73–87.

Karhatsu, H., Ikonen, M., Kettunen, P., Fagerholm, F. and Abrahamsson, P. (2010) 'Building blocks for self-organizing software development teams a framework model and empirical pilot study', *2010 2nd International Conference on Software Technology and Engineering*, IEEE, Vol. 1, pp.V1–297.

Kisielnicki, J. (2014) 'Project management in research and development', *Foundations of Management*, Vol. 6, No. 3, pp.57–70.

Kostami, V. and Rajagopalan, S. (2014) 'Speed–quality trade-offs in a dynamic model', *Manufacturing & Service Operations Management*, Vol. 16, No. 1, pp.104–118.

Lee, I. and Lee, K. (2015) 'The internet of things (IoT): applications, investments, and challenges for enterprises', *Business Horizons*, Vol. 58, No. 4, pp.431–440.

Martilla, J.A. and James, J.C. (1977) 'Importance-performance analysis', *Journal of Marketing*, Vol. 41, No. 1, pp.77–79.

Martin, A.S.A. and Whitworth, X.W.E. (2017) 'Agile processes in software engineering and extreme programming', *11th International Conference, XP 2010, Proceedings*, Springer, Trondheim, Norway, 1–4 June 2010, Vol. 283, pp.151–166.

Mascitelli, R. (2007) 'The lean product development guidebook: everything your design team needs to improve efficiency and slash time-to-market', *Technology Perspectives*, pp.3–199.

Matawale, C.R., Datta, S. and Mahapatra, S.S. (2016) 'A fuzzy embedded legality assessment module in supply chain', *Benchmarking: An International Journal*, Vol. 23, No. 7, pp.1937–1982, https://doi.org/10.1108/BIJ-12-2013-0113.

Meier, C., Yassine, A.A., Browning, T.R. and Walter, U. (2016) 'Optimizing time-cost trade-offs in product development projects with a multi-objective evolutionary algorithm', *Research in Engineering Design*, Vol. 27, No. 4, pp.347–366.

Menon, S. and Suresh, M. (2020) 'Organizational agility assessment for higher education institution', *Journal of Research on the Lepidoptera*, Vol. 51, No. 1, pp.561–573.

Mirza, M.S. and Datta, S. (2019) 'Strengths and weakness of traditional and agile processes – a systematic review', *JSW*, Vol. 14, No. 5, pp.209–219.

Müller, R.M. and Thoring, K. (2012) 'Design thinking vs. lean startup: a comparison of two user-driven innovation strategies', *Leading through Design*, Vol. 151, pp.91–106.

Paasivaara, M. and Lassenius, C. (2016) 'Challenges and success factors for large-scale agile transformations: a research proposal and a pilot study', in *Proceedings of the Scientific Workshop Proceedings of XP2016*, pp.1–5.

Paasivaara, M., Behm, B., Lassenius, C. and Hallikainen, M. (2018) 'Large-scale agile transformation at Ericsson: a case study', *Empirical Software Engineering*, Vol. 23, No. 5, pp.2550–2596.

Patil, M. and Suresh, M. (2019) 'Modelling the enablers of workforce agility in IoT projects: a TISM approach', *Global Journal of Flexible Systems Management*, Vol. 20, No. 2, pp.157–175.

Patil, M.R. and Suresh, M. (2021) 'Assessment of team agility in internet of things projects', *Webology*, Vol. 18, No. SI01, pp.137–148.

Perks, H. (2004) 'Exploring processes of resource exchange and co-creation in strategic partnering for new product development', *International Journal of Innovation Management*, Vol. 8, No. 1, pp.37–61.

Potdar, P.K., Routroy, S. and Behera, A. (2017) 'Agile manufacturing: a systematic review of literature and implications for future research', *Benchmarking: An International Journal*, Vol. 24, No. 7, pp.2022–2048, https://doi.org/10.1108/BIJ-06-2016-0100.

Rabiser, D., Grünbacher, P., Prähofer, H. and Angerer, F. (2016) 'A prototype-based approach for managing clones in clone-and-own product lines', in *Proceedings of the 20th International Systems and Software Product Line Conference*, pp.35–44.

Rabiser, D., Prähofer, H., Grünbacher, P., Petruzelka, M., Eder, K., Angerer, F. and Grimmer, A. (2018) 'Multi-purpose, multi-level feature modeling of large-scale industrial software systems', *Software & Systems Modeling*, Vol. 17, No. 3, pp.913–938.

Rajan, R. and Dhir, S. (2020) 'Technology management for innovation in organizations: an argumentation-based modified TISM approach', *Benchmarking: An International Journal*, Vol. 28, No. 6, pp.1959–1986, https://doi.org/10.1108/BIJ-01-2020-0019.

Rane, S.B. and Narvel, Y.A.M. (2021) 'Re-designing the business organization using disruptive innovations based on blockchain-IoT integrated architecture for improving agility in future Industry 4.0', *Benchmarking: An International Journal*, Vol. 28, No. 5, pp.1883–1908.

Raschke, R.L. (2010) 'Process-based view of agility: the value contribution of IT and the effects on process outcomes', *International Journal of Accounting Information Systems*, Vol. 11, No. 4, pp.297–313.

Reis, E. (2011) *The Lean Startup*, Crown Business, New York, Vol. 27.

Rigby, D.K., Sutherland, J. and Takeuchi, H. (2016) 'Embracing agile', *Harvard Business Review*, Vol. 94, No. 5, pp.40–50.

Roemer, T.A., Ahmadi, R. and Wang, R.H. (2000) 'Time-cost trade-offs in overlapped product development', *Operations Research*, Vol. 48, No. 6, pp.858–865.

Schulze, A. and Störmer, T. (2012) 'Lean product development–enabling management factors for waste elimination', *International Journal of Technology Management*, Vol. 57, Nos. 1/2/3, pp.71–91.

Shahin, M., Babar, M.A. and Zhu, L. (2017) 'Continuous integration, delivery and deployment: a systematic review on approaches, tools, challenges and practices', *IEEE Access*, Vol. 5, pp.3909–3943.

Sheffield, J. and Lemétayer, J. (2013) 'Factors associated with the software development agility of successful projects', *International Journal of Project Management*, Vol. 31, No. 3, pp.459–472.

Sindhwani, R. and Malhotra, V. (2017) 'A framework to enhance agile manufacturing system: a total interpretive structural modelling (TISM) approach', *Benchmarking: An International Journal*, Vol. 24, No. 2, pp.467–487, https://doi.org/10.1108/BIJ-09-2015-0092.

Sindhwani, R. and Malhotra, V. (2018) 'An integrated approach for implementation of agile manufacturing system in an Indian manufacturing industry', *Benchmarking: An International Journal*, Vol. 25, No. 4, pp.1106–1120, https://doi.org/10.1108/BIJ-01-2017-0017.

Sridharan, V. and Suresh, M. (2016) 'Environmental sustainability assessment using multigrade fuzzy – a case of two Indian colleges', *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, IEEE, pp.1–4.

Suresh, M., Yuvaprasanth, R., Nathan, R.A.R. and Amarnath, K. (2020) 'Employees stress level assessment: a case of apparel industry', *IOP Conference Series: Materials Science and Engineering*, Vol. 954, No. 1, p.012018.

Takeuchi, H. and Nonaka, I. (1986) 'The new new product development game', *Harvard Business Review*, Vol. 64, No. 1, pp.137–146.

Tanpure, G., Yadav, V., Jain, R. and Soni, G. (2021) 'Adoption of product lifecycle management in new product development: a case study of automotive organisation', *Benchmarking: An International Journal*, Vol. 29, No. 5, pp.1546–1561, https://doi.org/10.1108/BIJ-04-2021-0181.

Tirumala, S., Ali, S. and Babu, A. (2016) 'A hybrid agile model using SCRUM and feature driven development', *International Journal of Computer Applications*, Vol. 156, No. 5, pp.1–5.

Tripp, J.F. and Armstrong, D.J. (2018) 'Agile methodologies: organizational adoption motives, tailoring, and performance', *Journal of Computer Information Systems*, Vol. 58, No. 2, pp.170–179.

Tschimmel, K. (2012) 'Design thinking as an effective toolkit for innovation', in *ISPIM Conference Proceedings, The International Society for Professional Innovation Management (ISPIM)*, p.1.

Van Rossem, S., Sayadi, B., Roullet, L., Mimidis, A., Paolino, M., Veitch, P. and Ollora, E. (2018) 'A vision for the next generation platform-as-a-service', in *2018 IEEE 5G World Forum (5GWF)*, IEEE, pp.14–19.

Van Rossem, S., Tavernier, W., Colle, D., Pickavet, M. and Demeester, P. (2019) 'Profile-based resource allocation for virtualized network functions', in *IEEE Transactions on Network and Service Management*, Vol. 16, No. 4, pp.1374–1388, DOI: 10.1109/TNSM.2019.2943779.

Vimal, K.E.K., Vinodh, S. and Muralidharan, R. (2015) 'An approach for evaluation of process sustainability using multi-grade fuzzy method', *International Journal of Sustainable Engineering*, Vol. 8, No. 1, pp.40–54.

Vinodh, S. (2011) 'Assessment of sustainability using multi-grade fuzzy approach', *Clean Technologies and Environmental Policy*, Vol. 13, No. 3, pp.509–515.

Vinodh, S. and Aravindraj, S. (2015) 'Benchmarking agility assessment approaches: a case study', *Benchmarking: An International Journal*, Vol. 22, No. 1, pp.2–17.

Vinodh, S. and Chintha, S.K. (2011) 'Leanness assessment using multi-grade fuzzy approach', *International Journal of Production Research*, Vol. 49, No. 2, pp.431–445.

Yusuf, Y.Y., Sarhadi, M. and Gunasekaran, A. (1999) 'Agile manufacturing: The drivers, concepts and attributes', *International Journal of Production Economics*, Vol. 62, Nos. 1–2, pp.33–43.

Zhang, H. and Xing, F. (2010) 'Fuzzy-multi-objective particle swarm optimization for time–cost–quality trade-off in construction', *Automation in Construction*, Vol. 19, No. 8, pp.1067–1075.