Tampere University of Applied Sciences

Modernization of Software Delivery Operations

Engineering quality into software

Jani Nietula

MASTER'S THESIS
May 2020

Master of Engineering
Strategic Leadership of Technology-Based Business

ABSTRACT

Tampere University of Applied Sciences
Master's Degree Programme in Strategic Leadership of Technology-Based Business

NIETULA, JANI:
Modernization of Software Delivery Operations
Engineering quality into software

Master's thesis 86 pages, appendices 9 pages
May 2020

Today's business environment is growing rapidly with the ever-increasing scale of new business demands. Enterprises worldwide are going through the industrial revolution in digitalization, redefining the way how software products and solutions to customers are developed, qualified, and delivered. Customers requires enterprises to keep focus on engineering customer-oriented solutions which satisfies their needs and ensures a best user experience.

Every now and then organizations are required to streamline their software delivery operations. Due to increasing complexity of business environments they need to evaluate continuously their efficiency in the productivity and ensure faster high-quality software deliveries. Therefore, organizations are building a new foundation for a way of working practices to come more increasingly agile, collaborative, and flexible.

The purpose of this master thesis is to present the current state of software delivery operations at Fastems and how to plan a future state with improvements to modernize these operations. This thesis will analyze how enterprises worldwide are proceeding with their modernizations of software delivery operations. The gathered information presents how enterprises worldwide are facing and managing a challenge of engineering quality into software. This thesis describes how organizations could reach better circumstances, where quality is everyone's responsibility across organization.

Case studies shares experiences how enterprises worldwide are taking steps to scale up their practices with Software Quality Engineering and Lean-Agile concepts. These practices wants to be embraced into software development lifecycles and engineering practices. This thesis will describe how these transformations can be managed with change leadership throughout the organization. The empirical research of this thesis is based on the results of "State of Agile" survey at Fastems and elaborating also their Lean-Agile transformation journey.

This thesis evaluates how organizations could stay competitive and adapt their businesses at enterprise scale to be closer to the customer with their needs. Organizations are shifting to a transformation to take it as a journey which never ends.

Key words: software quality engineering, software quality assurance, devops, scaled agile, lean, agile transformation

TIIVISTELMÄ

Tampereen Ammattikorkeakoulu
Teknologiaosaamisen johtamisen ylempi tutkinto-ohjelma

NIETULA, JANI:
Ohjelmistotoimituksien modernisointi
Laadunvarmistus osaksi ohjelmistokehitystä

Opinnäytetyö 86 sivua, joista liitteitä 9 sivua
Toukokuu 2020

Nykypäivän liiketoimintaympäristö on muuttumassa nopeasti yrityksien jatkuvasti kasvavien vaatimuksien ympärillä. Yritykset kansainvälisesti kulkevat läpi teollista digitalisoinnin vallankumousta määritellen uudelleen sitä, miten ohjelmistotuotteita ja -ratkaisuja asiakkaille tulisi kehittää, laadullistaa ja toimittaa. Asiakkaat vaativat yrityksiä keskittymään enemmän asiakkaille suuntautuvien ratkaisujen valmistamiseen, joilla täytetään heidän tarpeensa ja varmistetaan paras käyttäjäkokemus.

Aika ajoin organisaatiot tehostavat toimintojaan ja arvioivat uudelleen tuotantonsa tehokkuutta. Organisaatiot rakentavat työtavoilleen ja käytännöilleen muutosjohtamisen kautta uuden perustan tullakseen ketterämmiksi, yhteistoiminnallisemmiksi ja joustavammiksi. Näin varmistetaan korkealaatuisten ohjelmistojen nopeammat toimitukset yhä monimutkaisemmassa liiketoimintaympäristössä.

Tämän opinnäytetyön tavoite on esitellä Fastems-yrityksen ohjelmistotoimituksien nykytilaa ja suunnitella heidän tulevaisuuden modernisoituja käytäntöjä ohjelmistotoimituksille. Tämä opinnäytetyö tarkastelee sitä, miten yritykset kansainvälisesti ovat parhaillaan modernisoimassa ohjelmistotoimintojaan.

Opinnäytetyö antaa lukijalle enemmän ymmärrystä siitä, miten kansainvälisesti yritykset kohtaavat ja hallitsevat laadullisten tavoitteiden rakentamista osaksi ohjelmistoja. Opinnäytetyö kuvaa myös sitä, miten organisaatiot voivat saavuttaa olosuhteet, joissa laadullinen vastuu pyritään vakiinnuttamaan organisaation kaikilla tasoilla.

Tapaustutkimuksien kautta opinnäytetyössä jaetaan kansainvälisten yrityksien kokemuksia laadullisten tavoitteiden uudistamisissa ja ketteryyden skaalautumisissa osana organisaatioidensa muutosjohtamista. Uudistuksilla halutaan varmistaa korkea laatutaso ohjelmistokehityksen elinkaaren jokaisessa vaiheessa. Opinnäytetyön empiirinen tutkimus perustuu suoritetun ”State of Agile”-kyselyn tuloksiin Fastems-yrityksessä. Opinnäytetyö käsittelee myös yrityksen valmistautumista Lean-Agile muutokseen.

Opinnäytetyössä analysoidaan lisäksi sitä, miten muutokset ovat tukeneet organisaatioita pysymään paremmin kilpailukykyisenä ja tarjoamaan laadukasta palvelua lähellä asiakasta. Organisaatioiden siirtyessä tähän isoon työkulttuurillisen muutokseen, sitä on pidettävä matkana, joka ei pääty koskaan.

Asiasanat: ohjelmiston laadunvarmistus, devops, skaalautuva ketteryys, ketterä muutos

CONTENTS

1 INTRODUCTION 7
1.1 Fastems Oy Ab 9
2 THEORETICAL BACKGROUND 11
2.1 Software Quality Engineering 11
2.2 Software Quality Assurance 15
2.2.1 Software Test Automation 16
2.2.2 Shift Left and Shift Right testing 18
2.3 DevOps 21
2.4 Agile at Scale 22
2.4.1 Scaled Agile Framework 23
2.5 Change Leadership in the digital transformation 32
2.5.1 Kotter's 8-Step Change Model 33
3 RESEARCH METHODOLOGY 37
3.1 Data gathering method 37
3.2 Data analysis method 39
4 CASE STUDIES 41
4.1 Software Quality Engineering Worldwide 41
4.1.1 Quality Engineering at Microsoft 42
4.1.2 Quality Engineering at Google 43
4.1.3 Quality Engineering at Salesforce.com 44
4.1.4 Quality Engineering at Facebook 45
4.2 Agile at Scale Worldwide 45
4.2.1 Scaled Agile at John Deere 50
4.2.2 Scaled Agile at Volvo Cars 51
5 IMPROVING QUALITY INTO SOFTWARE AT FASTEMS 54
5.1 Current state 54
5.2 Future state creation with software quality improvements 55
5.2.1 Improvements with Software Quality Engineering 56
5.2.2 Improvements to Software Delivery 58
6 SCALING AGILE AT FASTEMS 63
6.1 Current state 63
6.1.1 State of Agile Survey 63
6.1 Future state creation with Lean-Agile improvements 74
6.1.1 Improvements via Lean-Agile transformation program 74
6.1.2 Improvements to Lean-Agile change leadership 75
7 RESULTS 77

7.1 Steps forward with Software Quality Engineering 77
7.2 Steps forward with Lean-Agile..... 78
8 DISCUSSION 81
REFERENCES 83
APPENDICES..... 87
Appendix 1. State of Agile Survey 87

GLOSSARY

AI	Artificial Intelligence
API	Application Programming Interface
ART	Agile Release Train
ATDD	Acceptance Test-Driven Development
BDD	Behavior-Driven Development
CD	Continuous Development or Deployment
CEO	Chief Executive Officer
CE	Continuous Exploration
CI	Continuous Integration
CT	Continuous Testing
DevOps	Development and Operations
E2E	End-To-End
IT	Information Technology
M2M	Machine-To-Machine
MMS	Manufacturing Management Software
ML	Machine Learning
ROI	Return-On-Investment
SAFe	Scaled Agile Framework
SADM	Scaled Agile Delivery Methodology
SDLC	Software Development LifeCycle
SQA	Software Quality Assurance
SQE	Software Quality Engineering
TAMK	Tampere University of Applied Sciences
TDD	Test-Driven Development
UI	User Interface
WIP	Work-In-Process
XP	eXtreme Programming

1 INTRODUCTION

“Define Your Culture Before It Defines You”, this statement has been coming more relevant nowadays when digital transformations are increasingly experiencing across organizations in different industries. Organizations are defining their working culture again, building new ways of working practices. Digital transformations require to be properly planned and managed. If having difficulties with these, it could lead to obstacles to adopting new practices and other critical initiatives. In the worst case, having consequences which could lead to the loss of key employees and the weakening of stakeholder support.

Today’s rapidly growing business environment with the ever-increasing scale of new business demands requires organizations every now and then to streamline their operations, boost productivity and fast forward the deliveries. Organizations are adopting their work more to the Agile methods and utilizing it as a framework to further integrate their functions of business and IT. A modern software and hardware engineering, product management and delivery operations are required to keep up and face these business landscape challenges every day.

Buzzwords like Software Quality Engineering and Lean-Agile are coming a proven mindsets and principles for organizations. Empowering them to be more collaborative, cross-functional, innovative, automated, self-managed and productive. These buzzwords will also support organizations to revise their knowledge for improved performance, market share, and value of profitability.

The strategy of modern software development in organization’s digital transformations are shifting to a concept called Software Quality Engineering (or SQE) which combines well-known practices of Agile, DevOps and Software Quality Assurance. As organizations are expanding the role of Software Quality Engineering throughout the SDLC (Software Development LifeCycle), it will require leaders in the organization to react and transforming rapidly their workforces to work on this recently reformed SQE concept. Organization must then be prepared to plan and manage transformation properly, ensuring successful continuation of business.

A modern software development based on SQE practices is coming more and more result-oriented. Relentless pace of the variable business environment has created a need for the solution, where SQE way of software development plays important roles addressing customer needs and staying competitive against other competitors.

Due to the high failure rates and negative customer feedback with traditional software development methodologies requires changing the focus on SQE concept. The lesson learnt studies of failure rates forced organizations worldwide to invest and building quality in to ensure better customer satisfaction and Return-On-Investment (ROI) for their business. The renewing of customer expectations will allow organizations to focus shift-left and shift-right test activities. Also including DevOps test automation by transforming the previous Quality Assurance processes into one driven by Software Quality Engineering.

However, organizations globally have understood that any single set of Lean-Agile or SQE practices will not fit for all and hard to be utilized by everyone within an organization. At the end of the day, the key thing is that how organization finally uses these practices and getting business benefits with them. Every organization will have their own path of journey to get these working for their business by learning processes, principles, and practices with concept in question.

For the subject company of this thesis, Fastems Oy Ab has been practicing Lean, Agile, DevOps and Quality Assurance part of their Software Development LifeCycle for a few years prior to this thesis work. However, Fastems's software deliveries and its role in the factory automation business has been growing rapidly during last couple years. One of key reasons is that new business demands includes all the time more customized solutions with increased complexity. Latest versions of Fastems products and customers solutions are increasingly including more software and data management. This will require Fastems and its organizations from time to time to evaluate and update their practices part of Software Development LifeCycle. It continues to keep organization competitive and respond new business demands promptly.

The purpose of this thesis is to describe the current state of software delivery practices at Fastems and create improved proposals for the future state how Fastems could modernize these practices more with Software Quality Engineering and Lean-Agile concepts. This thesis also analyses with several case studies how enterprises worldwide are modernizing their software delivery practices with these same concepts and what Fastems could learn from these experiences.

1.1 Fastems Oy Ab

Fastems Oy Ab (hereinafter Fastems) is a family owned company headquartered in Tampere, employing approximately 500 people worldwide having offices in Europe, North America, and Asia. Fastems is an internationally renowned manufacturer of factory automation systems. Its mission is to build a world where manufacturing is essential for the sustainable well-being of people. Offering solutions to improve customers' productivity and profitability. Over 4000 systems have been already installed and in use for international customers working in industries like Aerospace, Automotive, Production Technology and Job Shops.

Fastems offers solutions for the intelligent factory automation and software such as Flexible Manufacturing System, Part Handling Solution and Manufacturing Management Software.

FIGURE 1. Fastems MMS Integrates your Production (Fastems, 2019b).

Manufacturing Management Software (MMS) launched in 1982 and currently can be seen one of the most advanced production planning and execution software in the factory automation industry. The modern manufacturing environment requires seamless team-playing between integrated production data systems and machineries, with MMS solution manufactures are able to integrate individual data systems connecting the production data and resources together. Fastems has become considerable also as a software company in the factory automation industry with over 100 software developers and engineers already (Fastems, 2019a).

The continuous development of Manufacturing Management Software ensures its strong position also in the future to serve manufactures from small subcontracting companies to large aerospace enterprises. MMS provides for the manufactures possibility to update their production continuously without major interruptions and to lead their markets by perfecting quality, delivery times and costs. It also enables manufactures to get rid of manual and repetitive work of inputting production data by controlling production machinery and data systems from a single source (Fastems, 2019b).

2 THEORETICAL BACKGROUND

This chapter will introduce theoretical backgrounds for the main concept and its practices in this thesis work. At first, giving the introduction for the Software Quality Engineering concept in general which is still relative new concept for many organizations. The concept itself in the global IT business has been known already over last two decades including lot of books, publications, articles, and stories from enterprises about their journeys with this concept. Nowadays, this concept is getting more awareness also in the factory automation industry without speaking its visibility already in the software engineering domain worldwide.

This chapter will give then also introduction to the better-known practices of Software Quality Assurance, DevOps and Agile which builds the cornerstones for the Software Quality Engineering concept. However, decided to keep introduction of these practices in a level to share latest trends of them and how these practices could jointly support Fastems in a best way.

2.1 Software Quality Engineering

Quality is raising a clear need for better ownership to be managed, where it is coming to be everyone's responsibility across organizations. Organizations have understood that assuring the software quality is not a separate practice of software development and quality is not equal to software testing. Organizations are mixing their internal quality processes to integrate Software Quality Engineering concept into their software development and engineering practices. They are integrating test organizations part of development teams, shifting left and right test activities having stronger quality control through the SDLC.

Customer satisfaction requires enterprises to be focused and conscious shifting their software engineering operations from creation an engineering solution toward satisfying the customers. Providing ever more functionality with the improved end user experience reflects to the trend an increasing business-related recognition of the importance of software quality. Customers do not want to know

so much about bits and bytes. They want to a solution which satisfies their needs and ensures best end user experience with characteristics such as ease-of-use, security, stability, and reliability. The critical here is “satisfaction” which covers both functional and quality perception of the software solution being used (Suryn, W. 2014. Software Quality Engineering: A Practitioner's Approach. Page 1).

Quality engineering implements the collaborative and seamless E2E (End-to-End) process between independent organizations integrating methods and tools (Figure 2).

FIGURE 2. A diagram of quality engineering – E2E software quality (Quality Engineering. 2020).

Customers’ expectations in software solutions and products to have a certain maturity level is increasing to the level, where software quality turns to be a competitive advantage for the organizations in today’s rapidly growing business environment. This is driving organizations to be more innovative in software quality management together with product innovation. Next generation software solutions and products are designed to use more decentralized, ranging from services of mobile, sensors and cloud to the systems of cyberphysical and M2M (Machine-to-Machine). Multinational development teams which are geographically distributed is came to be a norm in many organizations. The ways of working in these team models is adding quality challenges to collaborate efficiently.

Organizations and software professionals globally are talking more about the software engineering to be considered as an E2E approach to quality management. The software quality management must be treated to be more integrated context than what it has been in the past to help development teams cope with various business demands and technical issues. The modern quality engineering is going beyond the traditional disciplines of software engineering, product management and IT management, integrating organization's business and IT strategies, risk management, business process views and so on (Breu R, Kuntzmann-Combelles A, Felderer M. 2014).

Previously, organizations have added their effort to validate the product quality in their last stages of SDLC which do not give so much possibilities and time to react possible changes to be made or fixing errors found during test validations. Software Quality Engineering gives great opportunity for organizations to take quality account in every stage of SDLC from the first planning stage to the last deployment stage. SQE ensures organizations to receive constant feedback of their product quality. The evolution of Information Technology with new intelligent software tools and practices are improving the overall product quality, but still end users are facing risky, unreliable, and unintelligent products wasting his or her time or money.

The purpose of quality engineering for the software, systems, and related services is to assist developers to build good, intelligent, and reliable products to meet high quality end user satisfaction for those who want to use software as easily as they use a dishwasher, to shield against faulty products and unprofessional suppliers. (Suryn, W. 2014. Page 2).

Witold Suryn (Suryn, W. 2014. Page 13) describes economic ramifications of Software Quality Engineering with the few facts from life when undertaking the challenge of engineering quality into software:

- Everything in software engineering boils down to the user's satisfaction
- Satisfaction is conditional to the overall behavior of the system, with software products in the first place
- The behavior of any software product is perceived through features and quality

- Features and quality of software product are expressed through requirements
- Any behavior-related requirement for software product may only be realized through code

These facts run to the conclusion that in most development projects, functionality and quality are natural enemies when talking about financial ramifications of engineering quality into software or system. There are very rare situations where the project budget is open; in all other cases, the budget defines the battlefield where functionality and quality fight for an upper hand (Figure 3).

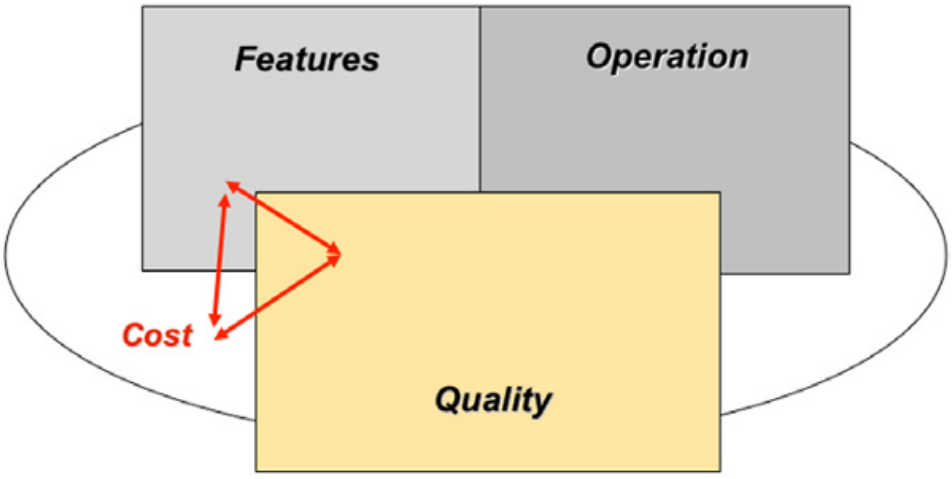


FIGURE 3. Functionality–quality battlefield. (Suryn, W. 2014. Software Quality Engineering: A Practitioner's Approach).

The Software Quality Engineering concept can be summarized to the following key enablers to support successful transformation:

- Increasing role of Quality Assurance confirming organization's engineering outputs
- Creating new Quality Engineering test capabilities supporting true shift left and shift right testing
- Adapting organizations to support the new engineering models to combine Agile, DevOps and Software Quality Assurance.

"The quality engineering process forms an integral part of the overall software engineering process, where other concerns, such as cost and schedule, are also

considered and managed” (Tian, J. 2005. Software Quality Engineering: Testing, Quality Assurance, and Quantifiable Improvement. Page 59).

2.2 Software Quality Assurance

When describing the software quality, it essentially is referring to the quality assurance within software engineering. It covers both internal and external quality. Any sold and provided software solution or product requires and emphasize the importance of quality assurance. External quality refers how the software is operating, considering for example performance and usability. Internal quality then specifies for developer how the code to be tested, maintained, and adjusted with new changes.

“Software Quality Assurance (SQA) is a systematic, planned set of actions necessary to provide adequate confidence that the software development process or the maintenance process of a software system product conforms to established functional technical requirements as well as with the managerial requirements of keeping the schedule and operating within the budgetary confines.” (Galin, D. 2004. Software Quality Assurance: From theory to implementation. Page 26).

Software Quality Assurance can minimize the cost of guaranteeing quality performed through the development and manufacturing phases. Quality assurance activities will prevent the causes of software defects and errors, giving then more time in advance to fix them in the early phase of development. With well-defined quality assurance practices within development lifecycle will reduce the rate of solutions or products that do not qualify for shipment (Galin, D. 2004. Software Quality Assurance: From theory to implementation. Page 29).

Over the last decade as the similar amount of software systems have been released to our everyday life, the number of bad news has been increasingly growing about problems caused by failures in software. Since in year 2008 in the Computer Weekly article Rebecca Thomson wrote about the Heathrow's Terminal 5 fiasco. The opening of Heathrow’s Terminal 5, technical problems with the baggage system caused more than 20.000 luggage’s misplaced and thousands of

passengers were left waiting for their bags. Due to these problems 20% of the flights had to be cancelled and British Airways lost over 15 million pounds. An investigation discovered that a lack of software quality was the main reason for the Terminal 5 fiasco, the solution did not include comprehensive software testing to identify gaps in different edge cases (Thomson, R. 2008).

Software technology revolution have caused organizations to react to the way how they develop, qualify, deliver, and operate their software solutions and products. We are talking about “Quality at Speed” where organizations must invest to be more innovative and capable to evaluate continuously their practices and tools to be able to develop and deliver high-quality software faster among the increasing complexity of business environments, decentralized services and data. (Top Software Testing Trends To Follow in 2020. 2019).

2.2.1 Software Test Automation

The demand of Quality at Speed in Software Quality Assurance drives organizations to adopt to the latest Agile and DevOps practices where test automation is nowadays playing a crucial role. Test automation helps teams to make tests more repeatable, reducing the risk for human errors, providing continuous feedback, and increasing test coverage. Test automation in organization’s Quality Assurance practices can save a huge amount of costs, time, and people effort, turning to be a competitive advantage in organization’s Software Development LifeCycles.

Nowadays, organizations are building their test automation infrastructure with a combination of open-source and commercial tools which they see that are providing the best approach for their development teams and supports organizations Agile practices as well as CI or CD (Continuous Development) methods. Software test automation professionals estimates that one of the test automation area which continues to grow in the future is the adoption of AI (Artificial Intelligence) and ML (Machine Learning) providing more augmentation or autonomous to the test automation (Herschmann & Murphy & Scheibmeir 2019).

Test automation plays a central position in organizations who are practicing Agile ensuring fast feedback to all its members. It enables testing capability and visibility to everyone who are working in the software development with Continuous Integration (CI) method. Keeping test automation working without interruptions increases organization's needs to invest to have specialized quality engineers to maintain the test automation infrastructure. The high use of test automation also makes the automated tests mission critical. If the test automation stops working or slows down, the development will also stop or slow down (Wiklund, K. & Eldh, S. & Sundmark, D. & Lundqvist, K. 2017).

According to Tricentis report (Tricentis. 2018) as illustrated in figure 4 manual software testing is still largely utilized in software organizations.

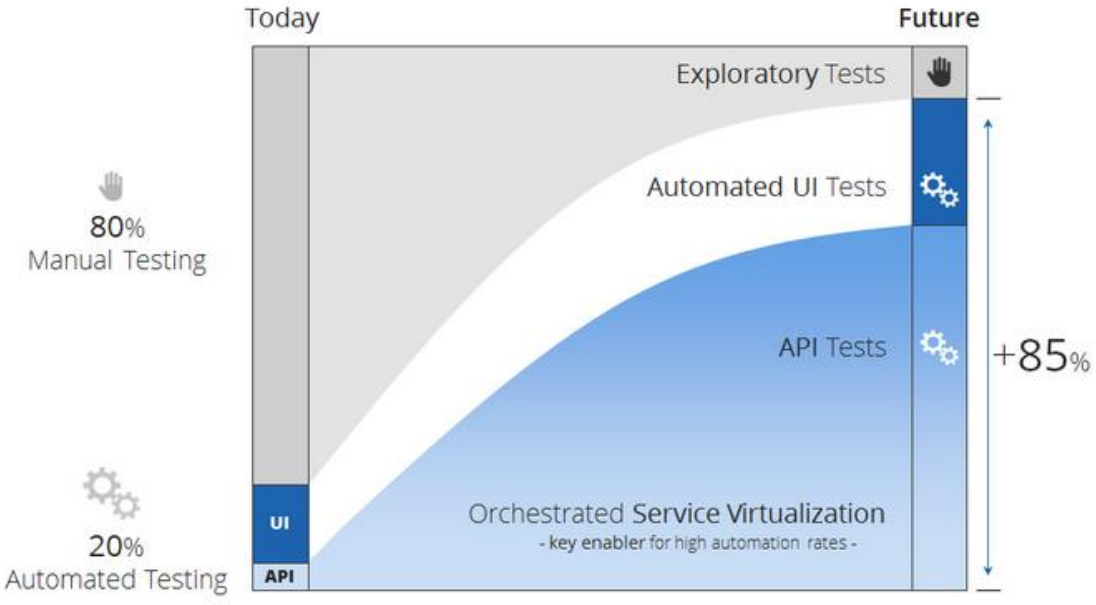


FIGURE 4. Why (and How) Manual Testing Must Change (Tricentis. 2018).

This creates a clear challenge transforming organizations to take steps towards Continuous Testing (CT) method with automated testing. Continues Testing is a method to run automated tests part of the organization's software development process. However, next steps in organization's the automated software testing requires also dramatically to be evolved from the UI test automation to the API test automation with better availability and accessibility via orchestrated service virtualizations. Without this evolution Continuous Testing with higher automated testing coverage will not be reality.

Test automation part of Software Development LifeCycle is one of the core software testing practices focusing to automate the validation of solutions and products against customer requirements. Software test automation tools enables development teams to automate such as unit, functional UI (User Interface), API (Application Programming Interface) and performance tests. Development teams can perform automated functional tests by driving the application UI or interact with the application through an API which gives more opportunities for development teams to start test automation in the very beginning. Test automation in successful DevOps practice is an essential part enabling continuous quality feedback loop.

2.2.2 Shift Left and Shift Right testing

Organizations globally have realized that software testing is important to be executed simultaneously within software development to maximize the benefits of Agile software development. 'Shift-Left' and 'Shift Right' testing are DevOps practices helping organizations to build quality into the software development projects right from the beginning. In the production phase addressing potential post-release problems before end-users are starting to phase them.

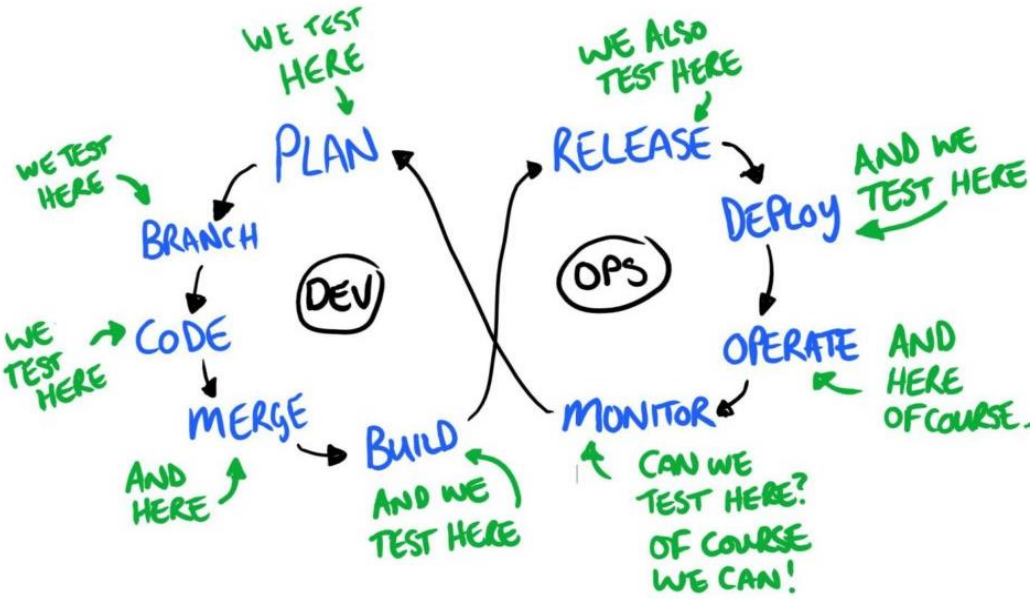


FIGURE 5. Continuous Testing (Blazemeter. 2019).

Embracing these two practices adds a great value to achieve real innovation throughout continuous software development and deliveries, combining software testing for development and operations to the Continuous Testing method.

'Shift-Left' testing is referring to the practice to start it as early as possible in the development process. It brings development and testing together. Nowadays, teams in Shift-Left testing are utilizing more techniques such as Test-Driven Development (TDD), Behavior-Driven Development (BDD) and Acceptance Test-Driven Development (ATDD) to incorporate testing into software development more seamlessly.

The driving idea behind shifting left helps to identify and fix defects more easily in the earlier phase of software development process. This enables now testing to happen every phase in the process. Instead of piling up majority of software testing tasks and activities till end of the process close to release date, teams are started to perform tests in smaller parts and more frequently. 'Shift-Left' testing helps teams to introduce better quality right from the beginning as well as saving time and resources.

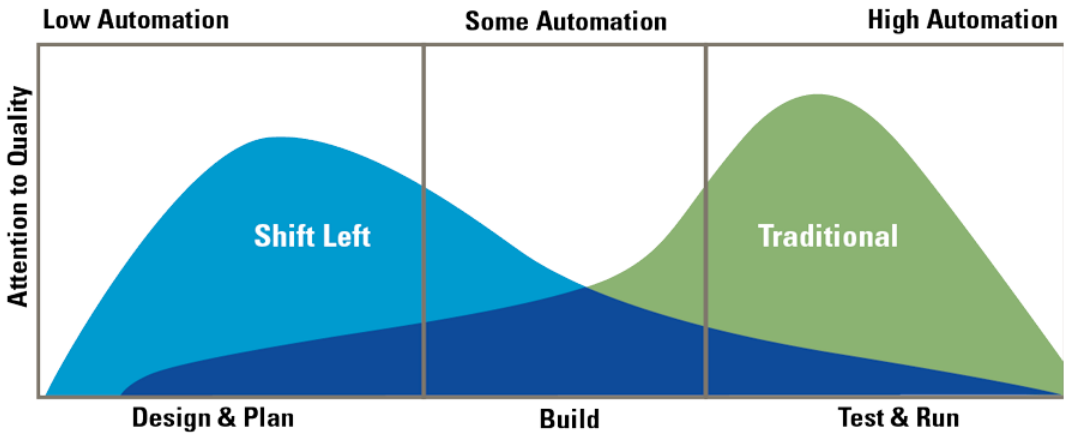


FIGURE 6. 'Shift-Left' testing mindset (Van Brummelen, J. & Slenders, T. 2019).

ATDD method is intended to create test cases before creating a code, representing expectations of the software behaviour to be implemented and delivered to customer. This method helps developers to understand the customer needs to be implemented and also allows customer to converse in their own domain language (Pugh, Kenneth. 2011).

Method of ATDD uses the Gherkin syntax, omnipresent Given-When-Then style to describe customer use cases close to natural language, lowering the bar for teams in the organization to start discussions and understand needs already from the very beginning.

This method highlights the collaboration model to be built between development, business and customer creating acceptance tests before code development starts.

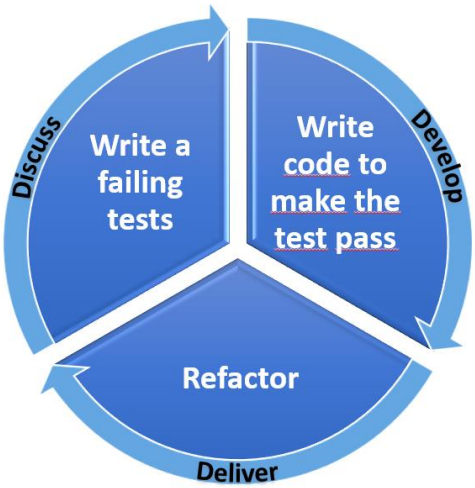


FIGURE 7. Illustration of TDD, BDD and ATDD approaches.

It involves people across organization from business, development and testing with different perspectives to define what needs to be implemented and how to ensure that it has been correctly. This collaborative discussion model is called “three amigos” (Agile Alliance. 2020):

- Business – What problem are we trying to solve?
- Development – How might we build a solution to solve that problem?
- Testing – What about this, what could possibly happen?

Andrew Grove, known also as a CEO of Intel said, “*How well we communicate is determined not by how well we say things, but how well we are understood*”. This quote summarises the key goal of utilizing ATDD method to build more collaborative specifications capturing knowledge and understanding of customer requirements allowing simple sharing across organization and not relying on memory of individuals or small amount of team members. This method increases better integration to software test automation, ensuring better quality of long-term documentation and maintenance services with customers.

‘Shift-Right’ testing is not yet so common as shift-left, but it starting to be more part of conversations, because only with shift-left practice it is not giving enough confidence to maintain performance, usability, and user experience. It brings operations and testing together, maximizing the test coverage. This practice helps to launch new features faster and test them by simulating behaviors in the production. Shift-Right testing could have a major impact to the teams how solutions are treated after release deployments to the production (Tozzi, Christopher. 2018).

2.3 DevOps

The concept of DevOps is getting more clarity in the organizations, what it is for us and how to use it efficiently in our business? DevOps can be understood that it is mainly about technical stuff with difficult buzzwords which can be hard to explain for a non-technical people. “People over process over tools” mantra gives much easier approach starting to understand the concept of DevOps. As many leading and most used Lean-Agile and Quality Engineering concepts and frameworks covered in this thesis starts with definition “what value it brings for the people?”, it also works with DevOps. Bridging people from separate software development and operations departments to the work in the same environment where developers and engineers collaborates throughout SDLC. This is how DevOps helps to remove the boundaries between different traditional development, Quality Assurance, and IT operation departments. Extending and enabling people more Agile way of working, advocating them for better communication and collaboration (Verona, J. 2016).

DevOps in practice, is expected making enterprise processes more efficient, reliable, and faster than what could be achieved with traditional software development methods. DevOps unifies practices from organizations that automates processes to be able to build, test and deliver solutions continuously faster and more reliably to customers.
DevOps tools and technologies without any doubt are the key drivers of DevOps. Tools for containerization, source control and continuous integration brings big

benefits in terms of automation and management for organizations to utilize DevOps (Thomas, N. 2018).

However, there are not a single set of tools which fits for all and can be used within an organization for the DevOps processes. The more important is that how organization uses these tools and getting benefits for own business. The concept of DevOps cannot provide one single path which works to every organization. Any DevOps playbook nor training cannot give straight answers for your organization how to get it implemented. DevOps journey would be best to start by learning its processes, principles, and practices. Taking pilots and learning from these is a one good approach to find a DevOps concept which brings most effective results to own organization.

DevOps and Agile continues to be the two most favored practices which businesses are using to stay ahead of the market. While integrating these two practices together organizations are started to get more benefits, experiencing better customer satisfaction and happier, more productive employees. Organizations must scale their DevOps and Agile beyond IT to be able to meet customer expectations with right products and quality. Advanced organizations are expanding DevOps and Agile throughout IT and outside of their development organizations. This will improve the velocity, greater acceleration, and ability to respond new opportunities with greater value (Cardoza, C. 2017).

2.4 Agile at Scale

Today, leaders in organizations globally are coming more familiar about the Agile team practices and benefits around it, working closer to customer to be able to serve them promptly and adapting to the changes. Leaders who are getting more experienced with Agile, are thinking forward their capabilities in own work community to scaling up Agile practices throughout the organization.

However, moving towards Agile at Scale in the organization, it requires to be done reasonably and realistically. In a way which fits best for own organization setup and business. All functions in the organization do not need to be organized into

Agile teams right from the beginning as these methods are not so easy to implement to all activities. It could not even be reasonable to implement with full coverage in the organization. More important is that functions who are not operating with Agile practices, they supports teams who do that (Rigby, D.K., Sutherland, J., Noble, A. 2018).

Why then organizations want to scale Agile? They want to stay competitive and adapting their businesses at enterprise scale to be closer to customer with their needs. They want to provide more flexibility to offered solutions, working in cross-functional team practices, and adapting common Agile ways of working also with other functions outside of software teams. Agile at Scale for organization is a cultural transformation committing people from different functions. Agile at Scale harmonizes practices and tools to improve collaboration, decision-making and transparency (Atlassian. 2019).

2.4.1 Scaled Agile Framework

Enterprises who are operating globally are actively searching the support how they could implement Agile methods successfully, changing how their people work to their global operations while organization continues to grow. Complexity arises when scaling Agile in globally distributed teams and projects due to global distance and a need of strong coordination among teams as well as within projects.

Scaled Agile Framework (SAFe) is one of the most adopted frameworks scaling Agile across the enterprise. Nowadays, SAFe is coming also more interested in small-to-medium-size enterprises and not just aiming for large-scale organizations getting similar benefits adopting Agile successfully. SAFe can been seen also as a container of several other existing Agile frameworks. It is having advantages in scalability and modularity, implementing Lean, Agile, and DevOps at scale (Razzak, M. & Richardson, I. & Noll, J. & Canna, C. & Beecham, S. 2018).

Scaled Agile Framework (SAFe®) founded by Dean Leffingwell who launched with group of Agile experts the version 1.0 of SAFe in 2011 including the first

training and certification program. During those days Dean and his team could not know that the Scaled Agile Framework would become the world's leading framework for enterprise agility, but as later described in this chapter it has been come also true (Scaled Agile. 2020).

The SAFe 4.6 version "SAFe® for Lean Enterprises" published in November 2018 focusing to increase success rates for the global enterprises and government agencies. It also adds more focus to the definition of Lean Enterprise and how they could increase successful deliveries of systems and solutions to their customers using Lean, Agile, and DevOps. The latest version 5.0 of SAFe was just released in December 2019, it is designed to expand enterprise SAFe with better achieved business agility to compete and thrive in the digital age (Scaled Agile. 2020).

On this thesis work, keeping focus in the SAFe 4.6 version as it sets the threshold for Fastems starting to embrace SAFe framework with certified trainers and leaders, utilizing the latest know-how and experiences from the field. SAFe® for Lean Enterprises provides a framework to be scaled and configured supporting company strategy and values in the best possible way. The version 4.6 of SAFe allows to refresh company's knowledge base for Lean, Agile, and DevOps with SAFe principles, practices, and competencies. Providing the guides to refine roles and responsibilities in a way to support organizational growth and activities to continue staying competitive against other competitors.

Potentially, this framework can increase better ways to align, collaborate and deliver with multiple Agile teams across the organization, improving productivity, quality, and employee engagements.

Figure 8 presents a visual overview of SAFe 4.6 framework "Big Picture", the center point of version 4.6 of SAFe is the five core competencies supporting to understand and implement SAFe for company's purposes.

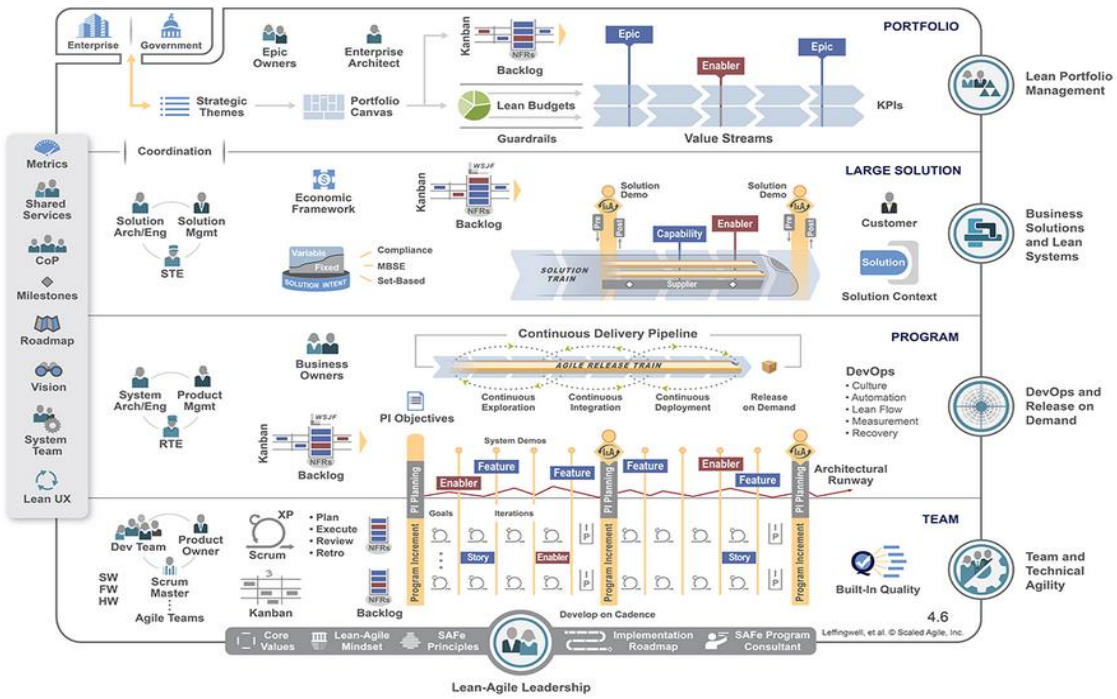


FIGURE 8. SAFe 4.6 Big Picture (Scaled Agile Framework. 2020).

The version 4.6 of SAFe represents the "*Five Core Competencies of the Lean Enterprise*": Lean-Agile Leadership, Team and Technical Agility, DevOps and Release on Demand, Business Solutions and Lean Systems Engineering, and Lean Portfolio Management.

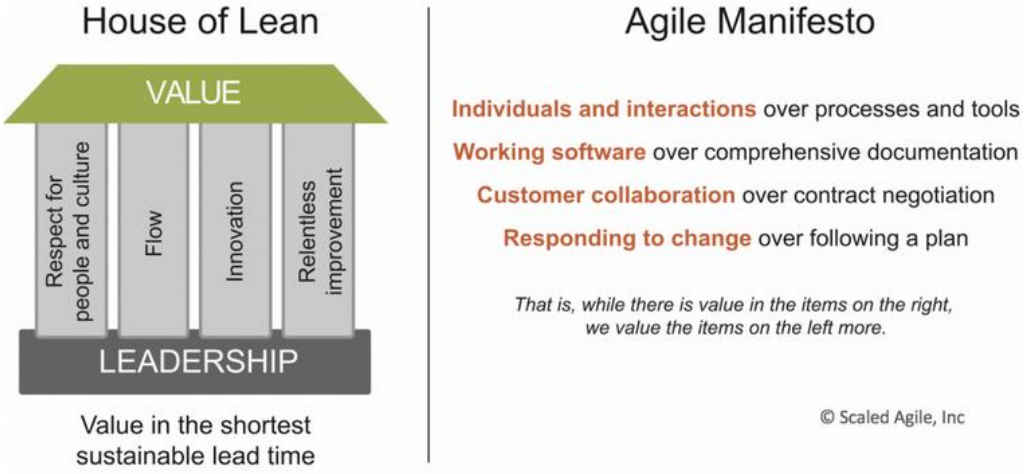
FIGURE 9. Five Core Competencies of the Lean Enterprise in SAFe 4.6 (Scaled Agile Framework. 2020).

Every core competency sets the main objectives for understanding and implementing SAFe, including together set of knowledge, skills and behaviors enabling

enterprises achieving higher quality and value in shortest sustainable lead time (Scaled Agile Framework. 2020).

1. **The Lean-Agile Leadership competency:**

Aiming for successful organization change, better employee engagement with increased productivity. Defines how Lean-Agile Leaders can empower their employees through the organizational change with leading by example, changing to



a new way of working and adopting a Lean-Agile mindset.

FIGURE 10. SAFe House of Lean and Agile Manifesto (Scaled Agile Framework. 2020).

The SAFe House of Lean demonstrates Lean thinking, where the base of the House of Lean represents Lean-Agile Leadership, creating the foundation for the four pillars. These four pillars "*respect for people and culture, flow, innovation, and relentless improvement*", exemplifies the goal of House of Lean: the Value. The value builds the deck on top of Lean-Agile Leadership with four pillars supporting the utmost goal to delivery highest customer value in the shortest sustainable lead time.

The Agile Manifesto provides the Agile principles and values for SAFe to be successful in Agile development activities. It is essential that Lean-Agile Leaders supports and emphasizes the importance of Agile Manifesto through the organization.

The Lean-Agile Leadership competency evolves the role of manager in a Lean enterprise. They are having a responsibility to lead their organization by exemplifying the core values, adopting a Lean-Agile mindset, and supporting the SAFe principles. They must do more than instruct, having a responsibility to guide their organization through the SAFe transformation; clarify each step in the transformation, what will happen in the next step and explain importance of why to keep going.

FIGURE 11. Lean-Agile Leadership is the anchor of the SAFe foundation (Scaled Agile Framework. 2020).

2. **The Team and Technical Agility competency:**

This competency is targeting to discover the critical skills and Lean-Agile principles and practices for Agile teams to be able to increase productivity, building quality in and delivering well-designed solutions with faster time-to-market.
The first part of competency is focusing to Team Agility, to create high-performing Lean-Agile teams who can master their team agility efficiently with Scrum and Kanban methods. Scrum method ensures better communication within the team. Kanban method helps then to optimize the flow by visualizing and managing teams flow of work.

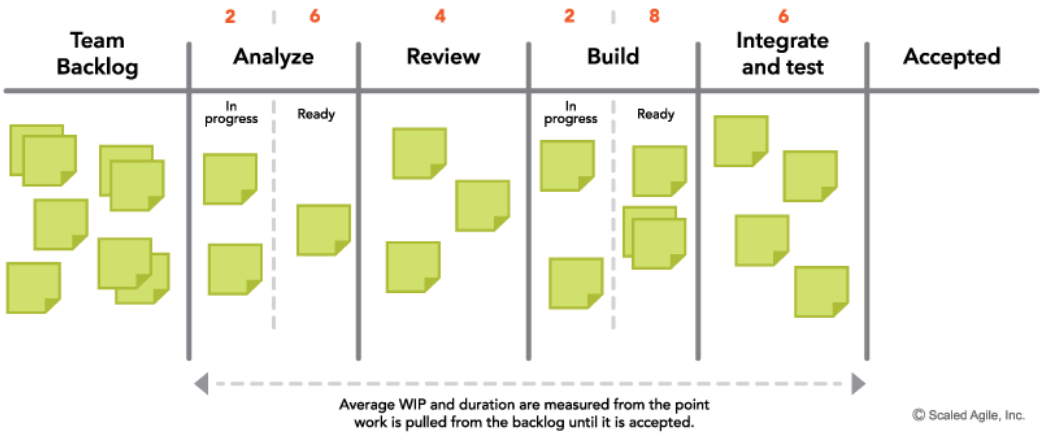


FIGURE 12. Lean-Agile Leadership is the anchor of the SAFe foundation (Scaled Agile Framework. 2020).

Visualization supports teams to identify potential bottlenecks with work-in-process (WIP) limits to not take new user stories under work before finishing ongoing story work.

The second part of this competency stands to Technical Agility, specifying the principles and practices for the Agile Software Engineering work to be able to delivery systems and solutions for customers in shortest sustainable lead time. Achieving a state of continuous value flow requires building quality into organizations Agile Software Engineering work. Technical Agility defines the Built-In Quality guidelines with the five key dimensions:

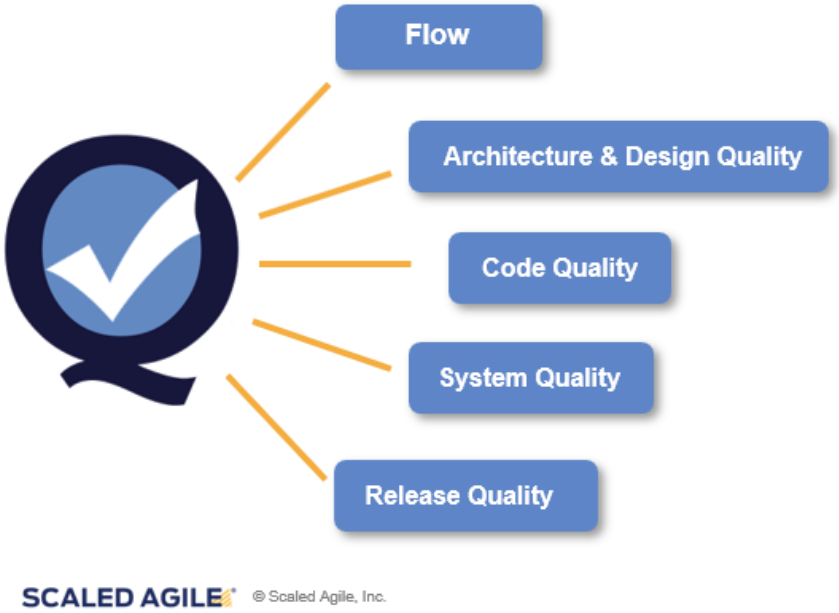


FIGURE 13. The Built-In Quality with five dimensions (Scaled Agile Framework. 2020).

- Establishing the **Flow**; giving a solid flow-based environment for Agile teams to develop and deploy deliveries faster and high-quality. This environment will support Agile teams continuously evolve latest and next business demands and requirements.
- Creating a solid object-oriented design principles and patterns for **Architecture and Design Quality** with well-defined interfaces.
- Encouraging eXtreme Programming (XP) practices to ensure good **Code Quality** with aligned coding standards, pair-work with continuous review process and collective ownership.

- Ensuring the best **System quality** creating the alignment and flow with 'Shift-Left' testing using methods like Test-Driven Development (TDD), Behavioral-Driven Development (BDD) and/or Acceptance Test-Driven Development (ATDD)which creates a collaborative approach between business and development teams with common understanding of customer requirements.
- Improving **Release quality** with implementation of Continuous Integration (CI) to keep all developers working with latest artifacts and Continuous Deployment (CD) providing capability to move faster towards production with latest updates.

3. **The DevOps and Release on Demand competency:**

Enables organization more towards customer-centric approach to define, build, and release solutions with highest value to delight customers better and fasters, in whole or in part, at any time to meet customer expectations. This competency describes the importance of releasing value to the customer when they need, building practices and mindset with Continuous Delivery Pipeline and DevOps. Depending organizations businesses, need for delivering continuously could be less important than the need to Release on Demand. However, DevOps and Release on Demand competency will bring competitive advantage to react customer demands more promptly by optimizing the collaboration between development and operations. Building better alignment of customer needs and requirements in the organization to increase more frequent deliveries to the customer.

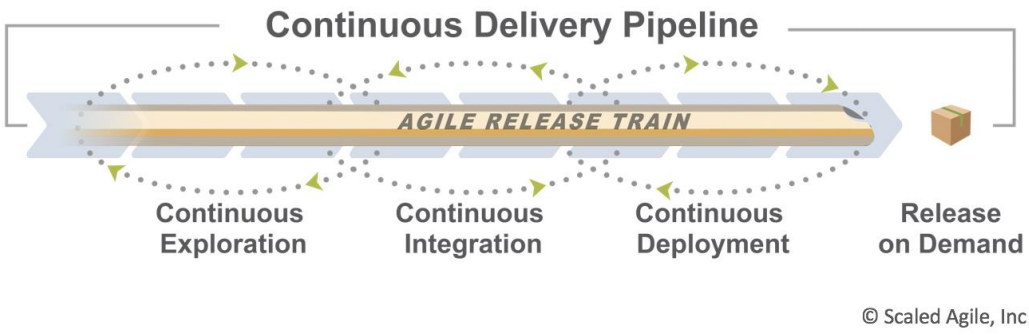


FIGURE 14. The Continuous Delivery Pipeline concept (Scaled Agile Framework. 2020).

- **Continuous Exploration:** helps to get forward with alignment of the customer demands and needs. Encouraging for more frequent customer visits designing and analyzing the solution which provides the best value for customer.
- **Continuous Integration:** builds quality into the Agile development lifecycle. Adding all work under version control and ensuring that new functionalities are built and verified end-to-end into a full system or solution before deploying them forward.
- **Continuous Deployment:** enables the process to take latest successfully verified and monitored changes from the staging environment and deploying them to production. Once new changes are in production, this step gives then also possibility determine are changes ready to be released officially to the customer. This step also allows possibility to withdraw latest changes or provide quick fixes when necessary.

4. **The Business Solutions and Lean Systems Engineering competency:**

Applies Lean-Agile principles and practices for organizations development lifecycle continuously delivering new solutions, capabilities together with latest technology upgrades. This competency feeds organizations to dream big and fosters them to adapt and invest to the large solutions with culture of innovation and learning. The following principles provides tools how to scale and produce customer solutions faster, enabling better predictability, fitting for the purpose with high-quality outcomes.

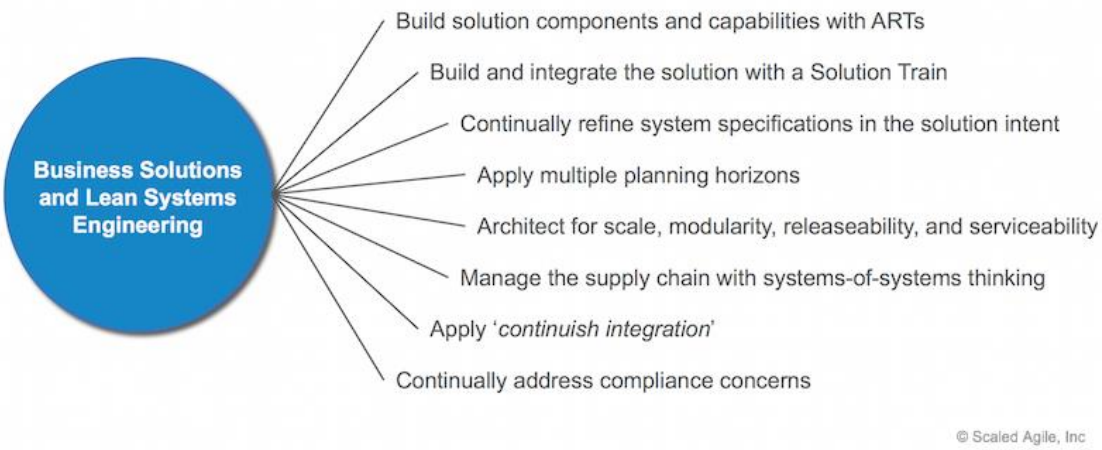


FIGURE 15. Key practices for building large solutions with SAFe (Scaled Agile Framework. 2020).

The Agile Release Train (ART) concept aiming for the long-lived team of Agile teams together with other stakeholders incrementally developing, delivering, and operating one and several solutions. ART coordinates and builds alignment for the groups of individuals (from 50 to 125 people) to the cross-functional Agile team, allowing to have all the capabilities as a Team-of-Agile teams to define, implement, test, deploy, release and operate solutions.

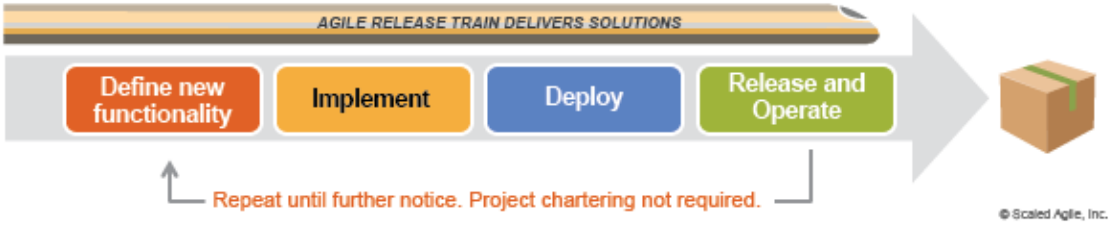


FIGURE 16. Agile Release Train concept (Scaled Agile Framework. 2020).

5. **The Lean Portfolio Management competency:**

Addressing organizations current portfolio management concerns and which solutions should be built and why. Helps organizations to modernize their portfolio management to support the new, Lean-Agile way of working. This competency helps organizations to connect the portfolio to the enterprise strategy and its execution with new Lean approaches to the budgeting and investment guardrails, managing portfolio operations and providing Lean governance model.
Organization which is not familiar with Agile software capitalization in portfolio management, it could delay transforming to the Lean-Agile practices in the organization. Therefore, organizations needs to modernize their portfolio management to support the new Lean-Agile ways of working.

Traditional Approach	Lean-Agile Approach
#1 Centralized control	Decentralized decision-making
#2 Project overload	Demand management; continuous value flow
#3 Detailed project plans	Lightweight, epic-only business cases
#4 Centralized annual planning	Decentralized, rolling-wave planning
#5 Work breakdown structure	Agile estimating and planning
#6 Project-based funding and control	Lean-Agile budgeting and self-managing Agile Release Trains
#7 Waterfall milestones	Objective, fact-based measures and milestones

FIGURE 17. Moving from traditional mindsets to the Lean-Agile thinking (Scaled Agile Framework. 2020).

2.5 Change Leadership in the digital transformation

Insanity: doing the same thing over and over again and expecting different results. - Albert Einstein

Often the word “change” is causing lot of different feelings, thoughts, and opinions. It could mean something good, negative, necessary, or difficult. Sometimes it could also mean something which would better to just delay to the future or avoid it completely.
People reacts different ways for changes and they can go through different stages as in figure 18 the Kübler-Ross Change Curve describes five stages to understand better how people are reacting to the change at different times. Some people approves or adapts to it very easily, but some people are against all kind of changes or it will take from them much more time to use to it.

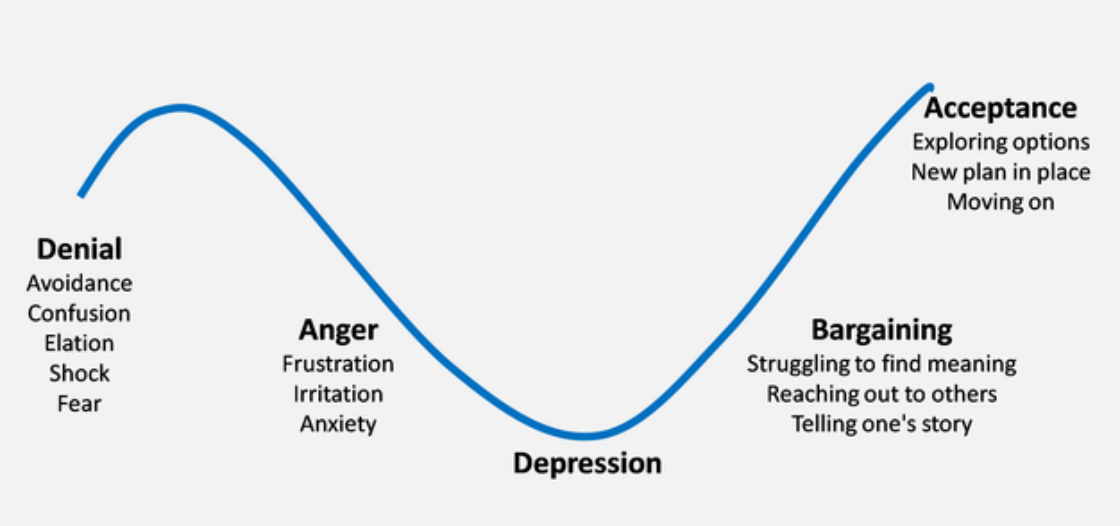


FIGURE 18. The Kübler-Ross Change Curve.

A term “Leading a change” or “Managing a change” shifts then discussions related to the change to the new level and adding different perspectives to understand it properly in different contexts that what it really means.

John Kotter (Kotter, J. 2011) presents in his article a difference between Change management and Change leadership. Kotter is very well-known expert of leadership and change management globally in this context. He states that terms

Change management and Change leadership are not interchangeable. A difference between these two terms is remarkable. Change management is more well-known term and keeping possible distraction and impact of the change as minimum level as possible. Managing the change with set of basic tools or structures under control with less effort.

Change leadership is then more comprehensive regarding large-scale transformations to drive people into new way of working or sharing and implementing a new vision. Change leadership is more about creating something totally nontraditional. It aims to lead masses of people be modernized and transformed to the new way of working, who want to make something happen and empowering them for the change (Kotter, J. 2011).

2.5.1 Kotter's 8-Step Change Model

A book “Leading Change” was published in 1996 by John Kotter, introducing eight-step change process. In 2014, he introduced to the enhanced 8 Steps, also known as the 8 Accelerators. Kotter defines the renovated 8 Steps with accelerators in a following way (Kotter International. 2020):

- **Create a sense of urgency,** around the need for change. Important factor for successful transformation is to build urgency first for the people having a window of opportunity, bringing them together, aligning commonality and clarifying steps forward.
- **Build a guiding coalition**, ensuring that people are receiving clear and combined information into new ways of working and that the change is necessary. This requires strong leading focus from leadership and key persons from the organization at all levels, managing the change is not enough.
- **Form a strategic vision and initiatives**, making vision of change become reality it requires strategic initiatives to be targeted and coordinated. A vision of change must be communicable, feasible, simple, flexible, and imaginable.

- **Enlist a volunteer army**, for the large-scale transformation it is important to have significant amount of people collectively sharing a common target and driving it to the same direction.
- **Enable action by removing barriers**, identifying people who could resist the change and help them to embrace the change. Also, inefficient processes or practices could require removing obstacles to execute the vision of change forward.
- **Generate short-term wins**, feeding your organization with quick wins to be collected and communicated early and often. A win (small or big) is the key motivator to help going forward with the change.
- **Sustain acceleration**, ensuring that the change is not announced too early after few wins. Requires leadership and key persons pressing harder the change forward until the vision of change is come a reality.
- **Institute change**, anchoring the change into organization culture. Cultural changes requires to be deep into a transformation and new behaviors are continuously repeated over the long-term.

FIGURE 19. Kotter's 8-Step Change Model (Kotter International. 2020).

The first seven accelerators are the ones which are constructing new capabilities, behaviors, and ways of working for the organization. The last one, accelerator number eight is maintaining them to the future. A key challenge in large-scale transformations are that how well and deeply new practices are rooted and anchored to the organization to be able to replace old behaviors and practices.

Tuula Niemi (Niemi, T. 2019) presented in the Technology Leadership course key themes in today's leadership transformations:

- Changes in the enterprises are constant, this requires organizations taking it as a journey which never ends.
- It is easier to explain the need for change when enterprise is going through the challenging times.
- The change requires that leadership together with full workforces will take an active role to implement the change successfully.
- Change leadership at all levels in the organization is important. Your closest superior is the most important leader in the transformation.
- Fears, dangers, wishes, and opportunities are essential part of the change. As a supervisor or leader, dealing these an asset with the workforces could lead successful progress in the transformation.
- Enterprises having ability and readiness for transformations can see it as a competitive advantage.
- Challenge of change leadership is to find right methods to motivate people and develop interaction part of the transformation.

The course included also one of the change leadership articles from Talouselämä magazine (Talouselämä. 2018) where transformations must move from the fact-based more to human-based approach.

Transformation must appeal to people feelings, focusing to the people who implementing the vision of change and bringing it to reality. Good leaders are able to take account people feelings part of the transformation and turn it to the important asset while implementing the changes.

Article included three key items to keep in mind in the successful large-scale transformation:

1. The transformation is defined in a way that it will evoke feelings in people, meaning curiosity, passion, confusion or even irritation. The transformation cannot be no-brainer or qualityless. People part of the transformation must have space for innovation and possibility to set their goals supporting the transformation in a best possible way.
2. The transformation includes a team where prevails a psychological trust. It means that people in the team dare to share their information, thoughts, feelings, and views without fare of any negative consequences. It will build a trusted collaboration within a team and they have then excellent chance to reach emotional state to allow openness and trustworthiness in a team towards successful transformation.
3. The transformation follows the progress of transformation continuously and sharing information to all who are part of the transformation. Creating visibility to the relevant transformation data, so that information is available for everyone.

3 RESEARCH METHODOLOGY

Research method in this thesis focuses to analyze quantitative data which is collected from the selected global researcher reports and survey analyses related to the current trends of Software Quality Engineering and Agile at Scale in the global organizations. Collected reports and surveys are giving an overview about pivotal pain spots of engineering quality into software and tackling collaboration challenges in Agile transformation programs.

Empirical research part agreed to use a survey as a method of gathering data. A quantitative survey is based on observations and measurements of the Lean-Agile transformation phenomenon gathered from the subject company of this thesis, Fastems.

An online survey as a research method is very commonly used and convenient to collect responses quickly from larger amount of people even, they are geographically dispersed. A survey can be frequently used to explain and explore different behaviors and as well as uncharted waters. A survey is a well-defined set of questions to which an individual is asked to respond. An online survey can be seen also as a cost-effective research method because data collection can be done fast, and analysis of data can happen right after survey is closed. Also, communication via online survey is convenient as internet is reducing the time and distance between people. It could include several question types and also different multimedia types photos, videos, and links to the support material. An online survey gives also more freedom for respondents to answer to a survey when the most suitable time for them via computer or handheld device (Walsh, T. 2016)

3.1 Data gathering method

An online survey was adapted based on the version 4.6 of SAFe Health Radar assessment approach monitoring the progress of organization in improving the flow of value through Lean-Agile transformation (Scaled Agile Framework. 2020).

An initial version of survey was introduced to the key stakeholders from the Fastems management. A survey reviewed and modified in cooperation to fit for the purpose in a best possible way. The content of survey agreed to be reviewed again after a first survey round and updating it based on the feedback, and key findings.

A survey aims to follow and observe the organization's status in the Lean-Agile transformation to understand better how well we are proceeding, where we are on this journey and which are the next steps to improve the journey. It focuses to measure the capabilities of the Lean-Agile organizations, guiding own organization to learn and implement them the most suitable way along the Lean-Agile transformation journey. A survey was built to measure and research continuously the current condition and progress in the different organization functions in question and also to be replicated or used by some other function in the organization. The feedback of survey itself is also collected in each round. All general comments and suggestions from the people are giving valuable information to improve survey for the next rounds.

An approved online survey includes in total 15 questions, starting with 2 clustering questions to gather relevant background information. An online survey ends to the open question which is optional to give general comments or suggestions about the survey, transformation topics etc. A survey was formalized to include the necessary explanatory material of each capability, to avoid possible misunderstandings and misinterpretations the statements of each capability was tried to keep as concrete and trivial level as possible. This also ensures organization to keep constantly Lean-Agile transformation related communication in-line ensuring that people are receiving clear and combined information into new ways of working.

An actual 12 survey questions measures the State of Agile looked at the capabilities of the Lean-Agile companies that we would like to develop at Fastems as well. These capabilities have relation to the SAFe 4.6 Lean Enterprise Core Competencies: Lean-Agile Leadership, Team and Technical Agility, DevOps and Release on Demand, and Business Solutions and Lean Systems Engineering

(Scaled Agile Framework. 2020). Lean Portfolio Management competency was not yet part of a first survey round and to be planned to a further survey content.

Each question is scored by using SAFe Health Radar scale ‘Sit, Crawl, Walk, Run, or Fly’ estimating where we are in our journey learning each capability. This scale helps organization to ascertain each problem areas based on its health radar to suggest next step to improve the flow and suggestion on what the next level or maturity requires from them. The health radar keeps focus more in the system level to concentrate to the capabilities which are good improve next and define actions to move for example from a ‘Walk’ to a ‘Run' (Scaled Agile Framework. 2020).

3.2 Data analysis method

Analysis of the first results completed with the quantitative data received from an online survey using the statistical methods. A survey data was analyzed first to look overall results summary and then visualizing the results with charts. Secondly, a survey data and general feedback was analyzed by key contributors in Lean-Agile transformation. A survey results of State of Agile are presented in chapter 6.1.1 via radar, pie, and horizontal bar charts.

A radar chart is also known as a spider chart, web chart or polar chart. This chart type is a good way to compare multiple quantitative variables and is useful to illustrate which variables are having similar values or if variables are outliers from each other. A radar chart gives a view how different variables are scoring high or low in a survey, illustrating the current performance or progress. However, it is better to keep radar charts simple and limit the number of variables. It helps charts to be more readable for the audience and not too complicate to get quick understanding of the results (FusionCharts. 2020).

A radar chart for a State of Agile survey illustrates how Lean-Agile capabilities are estimated per SAFe 4.6 Lean Enterprise Core Competence. Each capability can be examined quickly via radar chart and viewed how all levels look like. A

radar chart helps to identify what actions needs to be taken for each capability to get them to the next level.
A pie chart (or a circle chart) can be used to present the numerical proportion of each value (slice) to a total (pie). A pie chart for a State of Agile survey illustrates the respondent's role and functional group.

A number of horizontal bar charts for a State of Agile survey are visually presenting how each Lean-Agile capability are estimated by respondents in detail level. A horizontal bar chart is convenient way to manage a categorical data and compare bars in a same chart.

SAFe Health Radar instructs how to create a manageable plan to move towards next levels. Like creating a plan from 'Crawl' to 'Walk' across all capabilities than trying to get one capability to 'Fly' level as fast as possible. This keeps a plan balanced with full system view for all capabilities in own transformation journey and not trying to optimize just only one capability (Scaled Agile Framework. 2020).

4 CASE STUDIES

This chapter introduces and analyses real-life cases of Software Quality Engineering and Agile at Scale transformations from the enterprises working in the global organization models. This chapter shares an overview of successful method and practice changes, but also disclosing certain pivotal pain spots and potential bottle-necks to be avoided along with Software Quality Engineering and Agile at Scale transformations.

4.1 Software Quality Engineering Worldwide

Companies worldwide are upgrading their software development and engineering practices to come more increasingly agile, collaborative, and flexible. Companies are going through transformations to come more Agile as an organization. They have realized that software testing and quality with Agile and DevOps practices requires to be continually evaluated and promoted from the very beginning. Doing this as part of the software development process they can ensure the best possible customer experience.

Quality has become a critical attribute of software products as its absence produces financial, health, and sometimes life losses. At the same time the definition, or scope, of the domain of software quality has evolved continuously from a somewhat technical perspective to a perspective that embraces human aspects such as usability and satisfaction (Suryn, W. 2014. Software Quality Engineering: A Practitioner's Approach).

Figure 21 illustrates the Gartner report from 2018 which stated that “Ensuring Quality of Solutions or Features Developed” has been one of the main challenges in global organizations (Barnott, G. 2019). This same report also emphasizes importance of skill and competency development in the delivery organizations as well as efficient collaboration between business and development functions.

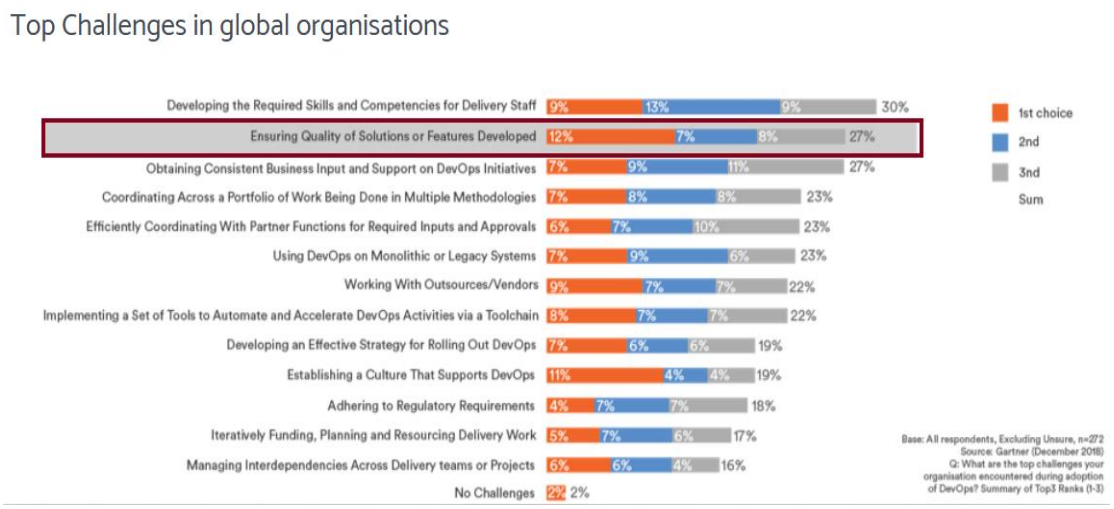


FIGURE 21. Top Challenges in global organizations (Barnott, G. 2019).

4.1.1 Quality Engineering at Microsoft

How Microsoft dragged its development practices from waterfalls into Agile have similarities which also many companies are going through before getting a pressure from the top to adopt these kinds of practices and become agile as an organization. Microsoft updated their engineering approach merging traditional developer and tester roles into one discipline: software engineers. Software engineers have responsibility for every aspect of making their features completed and performing well in production. Testing and quality became everyone's responsibility.

Microsoft's journey from waterfall development process into Agile was not the easy one. It is traditionally viewed as a company having series of fiefdoms, with each team jealously guarding its own work and not sharing with others. A different development teams have not had access to other development teams. They could not see what other teams were working on or the source code they were producing.

Microsoft started to move to the "combined engineering" Agile approach of combining development and testing. They started move programmatic test creation from testers to the developers. They do have still Quality Assurance and continuing to be important, but it is focusing more to the end-user style "real world"

testing and not programmatic automated testing. This move has been successful, improving the team's ability to ship changes without harming overall software quality (Bright, P. 2014).

Microsoft’s Agile Transformation took great effort and was anything but a straight path from A to B. They also understood that it is a journey and the journey never ends. Introducing the practices of Scrum were only part of the challenge, the bigger challenge was the mindset change for all involved. Teams got the autonomy to be the masters of the features what they will deliver to customers which enabled them to be more connected, understand and respond more efficiently to customer’s needs keeping them regularly delighted. This Microsoft’s industrial revolution engaged their teams to be more effective and operating faster with better quality (Denning, S. 2015).

4.1.2 Quality Engineering at Google

When your organization has over 2 billion lines of code and 25,000 engineers, how do you keep quality high? Google had to create a strategy for Quality Engineering to uphold demands of organizational needs to have teams more collaborative and shifting landscape of new features and products. Google realized that “Software Testing is a Focus, Not a Team”. They noticed that in their development team’s software engineers building and testing their own did not scale anymore. Due to this they created specialized engineering roles to ensure the organization’s QA bandwidth leveraging holistically without building traditional testing teams (Dotterweich, A. 2018).

“*Testing and development go hand in hand. Code a little and test what you built. Then code some more and test some more. Better yet, plan the tests while you code or even before. Test is not a separate practice; it is part and parcel of the development process itself. Quality is not equal to test; it is achieved by putting development and testing into a blender and mixing them until one is indistinguishable from the other.”* says James Whittaker, a former Microsoft architect and currently Director of Test Engineering at Google (Avram, A. 2011).

They consider quality to be a development issue, not a testing issue. That is why Google ensures quality as “an act of prevention” rather than one of detection. Google’s quality engineering roles are making sure that their developers have the requires test automation infrastructure and enabling processes to do Quality Assurance. Responsibility of code quality stays clearly in developers, utilizing created infrastructure and processes. According to Whittaker: “quality burden where it belongs: on the developers who are responsible for getting the product right” (Avram, A. 2011).

4.1.3 Quality Engineering at Salesforce.com

Salesforce has founded on a culture of quality and customer advocacy. Their Quality Engineering goes beyond testing. It unifies their development, product ownership, and user experience with the ultimate goal of best customer experience. Quality Engineering in Salesforce involves their organization to take part of decision making in every phase of development lifecycle where simple pass or fail test results will not make possible for you. Quality Engineering enables them to determine whether a product really meets customer needs, designed in a visually appealing, performant way (Baltierra, C. 2016).

Salesforce defined feature teams consisting of developers, Quality Engineering, doc writers, product managers and several other roles who were 100% dedicated to that specific team. They encourage quality engineers also to write feature code, like they also encourage their software developers automating functional tests, not just unit tests. Quality Assurance comes part of software development from the very beginning. It is built into the software development process having dedicated quality engineers on the team. They get rid of the division of “us vs. them” between quality engineers and developers. Salesforce have invested heavily to the extensive test automation infrastructure for most of their software solutions which allows them to release enhancements and updates with such frequency. The test automation infrastructure ensures also that their test and production environments to stay aligned which is solid base for the quick and frequent deployments.

Salesforce built the solid culture of quality and customer advocacy to their feature teams to ensure best possible customer experience which requires that quality and how it is assured is continually evaluated and promoted as part of the process from the very beginning. They all get involved already in the design process and the design reviews. Collaboration or as they called it partnership between quality engineers and developers where they began learning from each other. Quality engineers giving support for developers to grow and mature their quality mindset, and developers supports quality engineers to improve their software development skills (Meier, J. 2014).

4.1.4 Quality Engineering at Facebook

Facebook focuses on code ownership to make sure each developer is personally responsible for the quality of their own work. Facebook makes clear to software developers in job descriptions that Quality Assurance of software will be a part of the job utilizing their wide range of automated testing solutions.

A key to the Quality Engineering culture at Facebook is that developers have the responsibility for the code what they have written including proper Quality Assurance and giving support once it is in production. Facebook's code ownership model requires that developers also gives a support the operational use of their software, known as a DevOps practice.

This mode motivates developers to write code with high quality standards and running Quality Assurance continuously. The model builds strong ownership for developers to keep system running smoothly in production and emphasizing criticality of personal responsibility. It completes the culture of Facebook's Quality Engineering letting the system maintain quality at scale (Bird, J. 2013).

4.2 Agile at Scale Worldwide

Enterprises worldwide are looking new ways to operate which could influence more positively a company's speed and adaptability for the required change.

Agility at scale has been seen crucial for the enterprise businesses to be competitive in the digital age.

KPMG company concluded in 2019 the Survey on Agility (KPMG. 2019) for several enterprises with more than 120 participants from 17 countries. The survey results indicated that Agile is formalizing the baseline for the revolution of software development practices. Scaling agility across teams in the organization is coming to be a new normal, replacing outdated and traditional models in software development.

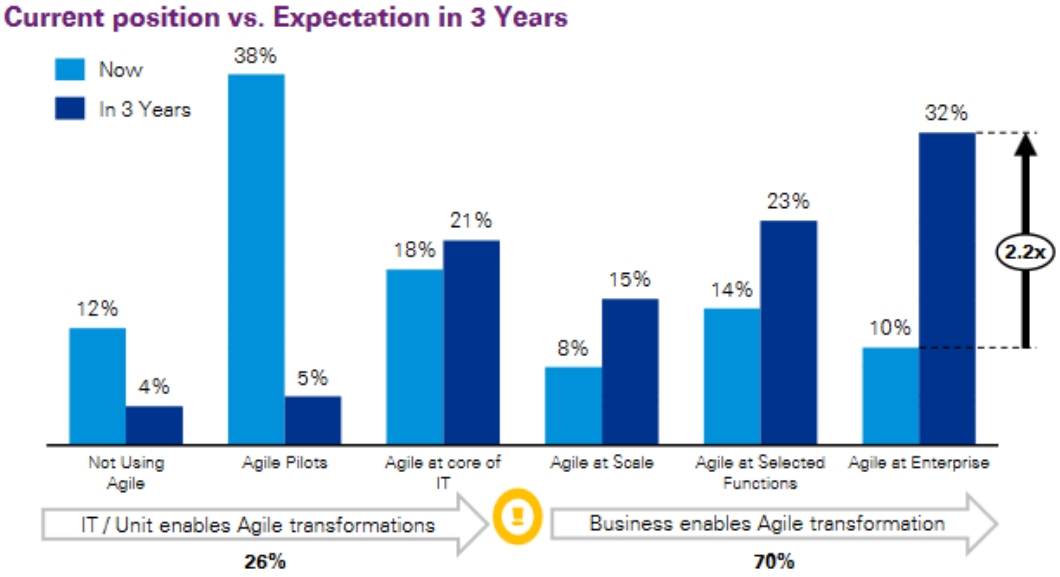


FIGURE 22. The KPMG Survey on Agility (KPMG. 2019).

Majority of the survey respondents reported that in the next three years their organizations have an ambition to proceed Agile transformation with Scaled Agile Framework (SAFe®) and successfully integrate their business and IT functions. Big part of organizations in this survey indicates that their key drivers for agility is to enable more speed to their product delivery with continuously improving customer satisfaction. Today's customers are expecting more from the deliveries and requiring organizations to have capabilities to respond faster to changing customer needs.

As illustrated in figure 23, the next key drivers to increase agility are related to a need to be more flexible across organization with Agile way of working, as well

as breaking down the silos between business and IT. Having a clear target to go forward with the enterprise strategy including both business and IT.

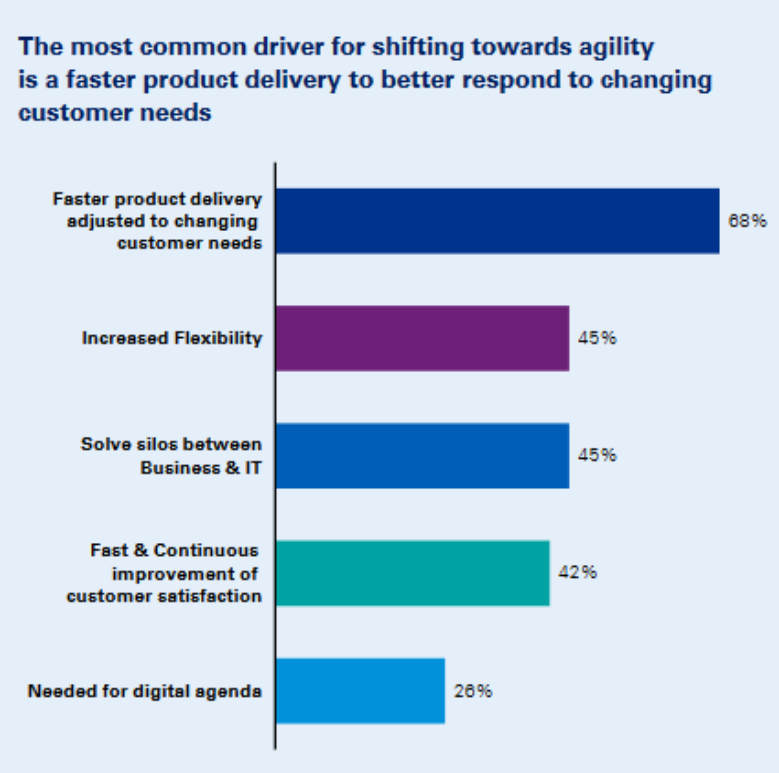


FIGURE 23. The KPMG Survey on Agility (KPMG. 2019).

The survey report emphasizes that successful Agile transformation requires strong leadership commitment to avoid model where transformation is tried to be managed and handled bottom-up which are usually ending to unsuccessful results. Together with strong leadership commitment another important part of successful Agile transformations is to ensure the trainings for leaders and teams are taken care to understand Agile principles. It has been seen also valuable to have coaching available in every day work to understand Agile behaviors at all levels in the organization from the team level to the executive board level.

Publication by Steven Zobel in 2018 (Zobel, S. 2018) reports about the challenges what enterprises are encountering in their digital transformations initiatives by investing globally more than one trillion US dollars in 2018. Apparently, based on McKinsey research 70% of these digital transformation initiatives are failing (McKinsey & Company. 2016). This is due to fact that enterprises have forgotten or not been able to build a coalition across the organization for the trans-

formation, hence missing the real teamwork driving required changes to the success. Organizations are missing the collaboration to communicate, coordinate and implement the change together. Also, this publication reports that enterprises are lacking an operation system to store, and record organization's investments, progress and results related to the digital transformation initiatives. This leads to the situation that centralized reporting about critical initiatives is not possible and that is why hidden.

Key takeaways from this publication are that leaders in the organization needs to understand and learn how their teams are actually delivering. Integrating their siloed teams to work as an enterprise team bringing visibility across the entire organization or product delivery lines. Establishing more automation to individual and cross-functional work. Centralizing data storage and progress reporting of ongoing work, consolidating the source of data.
At the end of the day, digital transformations is about the people and how they work. Enterprises must think carefully how they will treat their people through the digital transformation efficiently and successfully, applying a new modern way of working methods and concepts enabled by the new tools (Zobel, S. 2018).

Figure 24 presents the Gartner's enterprise Agile survey results from May 2019 to give more insights how chosen Agile Framework could maximize their potential in the organizations scaling Agile to their working culture and its challenges.

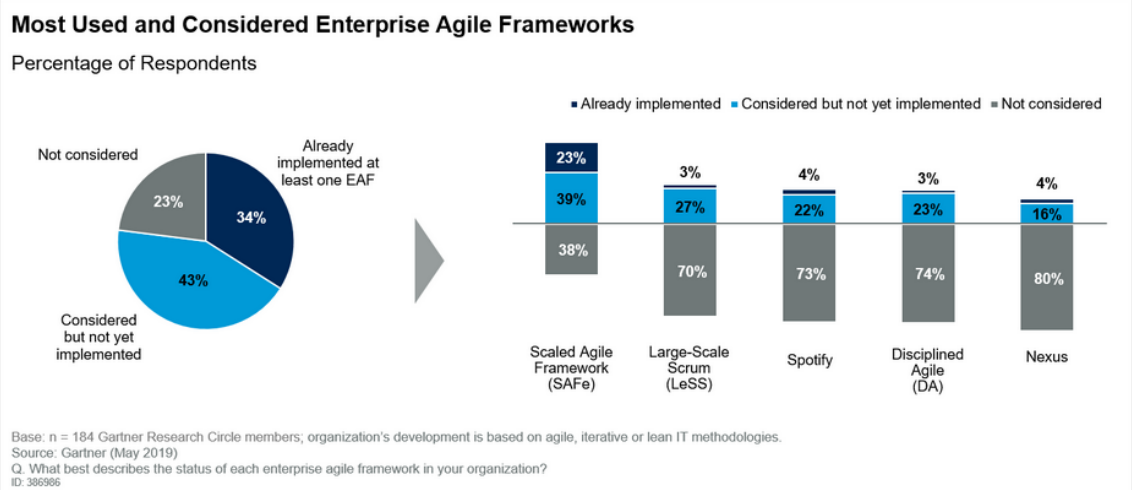


FIGURE 24. SAFe Is the Most Used and Considered Framework (Gartner. 2019).

The same survey results indicates that currently Scaled Agile Framework is the most implemented and considered framework, keeping frameworks like Large-Scale Scrum (LeSS) or Spotify far behind (Gartner. 2019).

How then SAFe framework could work also in large-scale government programs in the public sector organizations? Without any doubt it most probably includes also some unsuccessful stories or failed initiatives as described via couple reports above in this chapter, but having also some government agencies which are already successfully using SAFe, like Central Intelligence Agency (CIA), The Department of the Air Force (DAF) and Australian Post.

Natalie Field from the Australian Post says, "*SAFe has really helped bring the organization along its transformation journey. Its real value has been in the way it links strategy with decentralized execution, using metrics to enable a high level of transparency and fact-based decision making to focus on achieving business outcomes*". The Australian Post with more than 200 years history started to invest their technology, people, and culture some years ago, adding more focus to customer experience and continuous innovation.
They selected SAFe framework to help achieving these goals, not only for updating their operating model but using it as a tool in transformation. They set target for constant change which will fundamentally shifting how their organization approaching and delivering against their strategy. Building to the organization cross-functional teams with culture of curiosity, innovation, and learning. They adopted Agile Release Trains (ARTs) supporting their value streams and associated enterprise strategy. It created for them a customer-centric and continuous improvement culture with better business driven metrics (Scaled Agile Framework. 2020).

The world of Agile offers to organizations various frameworks and ways of working models how to scale Agile, but any model itself is not the whole truth and requires enterprise to act as a learning company to evaluate selected model and practices continuously. It requires organizations taking it as a journey which never ends.

4.2.1 Scaled Agile at John Deere

John Deere is one of the most notable companies with success story of SAFe transformations having systems which are embedded, real-time and complex. They decided to go forward with “all in” approach which impacted immediately about 800 software developers globally to the teams in US, Europe, and India. They also understood the software quality must be extremely high and ready on time due to fact that the delivery dates are fixed for new vehicles leaving from the factory always on time (Scaled Agile Framework. 2020).

John Deere managed to decrease their delivery times from 12-18 months to 2-4 weeks and increasing customer satisfaction and engagement work in their engineering teams. However, in those days 2011, they also realized that SAFe is not giving complete solution for addressing all digital transformation issues beyond the enterprise’s software development challenges (Bloomberg, J. 2014).

An article “Your Tractor Was Built With Agile” (InfoQ. 2012) gives a more insight about John Deere’s Agile transformation journey, taking first steps in 2007 when they started to experiencing Scrum method with few teams. However, in 2010 they realized that their capability to delivery new functionality is not possible without a substantial change in the way they are working. They started to remove silos between functions and organized their workforces into Agile teams, having a goal to spreading Agile practices to all functions in their global organization including their offshore locations as well as supplier network. They introduced SADM, or Scaled Agile Delivery Methodology combining Scrum, SAFe, XP and Lean practices. SADM provided for workforces in John Deere a clear guideline for common Agile way of working.

This big cultural change in John Deere faced the similarities as most of companies are going through in Agile transformation. They first underestimated the cultural change what they were undertaking. Basically, everyone’s roles and responsibilities were affected and took a time to reform. They continuously evaluated and evangelized their processes and business benefits to meet customer expectations. Also, John Deere knew that at the end the day, every enterprise needs to do all what they can to improve their processes to deliver the highest quality,

productivity, time to market and employee engagement or enterprise will simply fail to be successful. In the end of the article “Your Tractor Was Built With Agile” (InfoQ. 2012), John Deere shared their recommendation for enterprises who are looking possibilities to start their journey with Agile transformation: “*There is never a good time for significant change, and sometimes it seems easier to continue to do what you have always been doing. But the world is changing rapidly, and our customers and our market demands that we change too*.”

4.2.2 Scaled Agile at Volvo Cars

The article “why and how Volvo Cars embraces Agile at Scale” by Steve Denning (Denning, S. 2020), overviewing well Volvo Cars journey scaling their Agile implementation from the software community level to covering the whole product range, both hardware and software. Volvo Cars realized that cars are more and more becoming “computers on wheels” and they must change their methods which fits for this purpose. They had to get rid of development lifecycle where the hardware part, developing the physical car first as ready as possible and then later adding software.

To build “computers on wheels” required that they need change the way to develop both hardware and software simultaneously in an integrated fashion. They realized that they cannot develop some new features customizing them for every product, project, or car model. There was a need to start developing features in a way that can be utilized and re-used for other products, projects, and car models as well. This forced them to organize themselves to starting an initiative called “Agile Product Streams”. During those days in 2017 it was not an easy shift for the people to accept to move towards with Agile transformation in the complete product range and even today, they keep still reminding their people about necessity of Agile way of working at large scale.

Earlier initiatives to Agile practices before 2017 in Volvo Cars has been very similar than most of enterprises globally are going through. They had some individuals and teams in software development who has been interested about Agile and utilizing these practices in the small scale. However, in the company level they

were not able to scale it to cover the whole software community and not even thinking to have it in the larger scale covering complete product range. In 2017 top management at Volvo Cars together with their management team in software decided that they need a change scaling their Agile ways of working with Scaled Agile Framework (SAFe). From the very beginning they got a strong top management commitment and involvement in the transformation which built trust to move forward with this transformation.

The main objectives of the Volvo Cars Agile transformation were speed and responsiveness. Furthermore, they were also concerned about the quality, because one of their platforms was not in the acceptable level and forced them to start doing things in development in a different way. They had to start to modernize their management methods to be able to handle future needs customers. The part of Agile transformation Volvo Cars utilized thinking of Rogers Technology Curve for basically everything they wanted to do as presented in figure 25. They also used effort for highlighting the individuals or teams who are doing some good Agile practices (early majority).

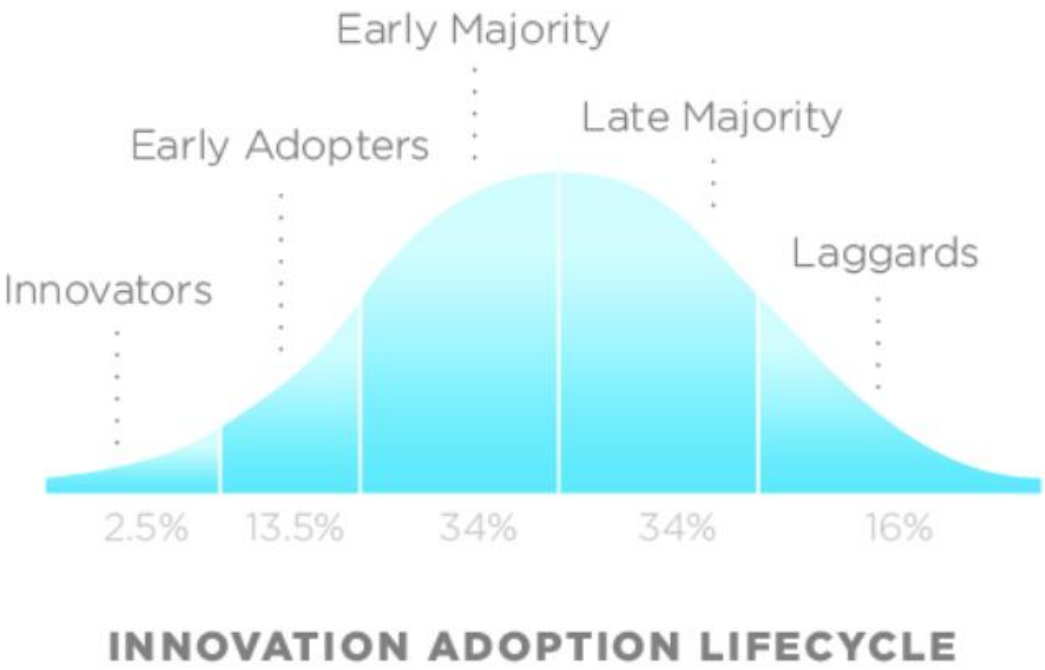


FIGURE 25. Rogers Technology Curve (Wikipedia. 2020).

All good learnings and practices including for example new role descriptions were stored to the Volvo Cars Agile Framework acting as an organizational memory. This gave then a base for others in the organization to implement and practice to their purpose (Denning, S. 2020).

Steve Denning's article about Volvo Cars (Denning, S. 2020) embracing Agile at Scale concludes that Volvo Cars have educated thousands of their people with intensive SAFe trainings and they have seen it important to get their people understand what SAFe really is and how it could work in practice. They see that to be able to get implementation of Agile at Scale on the right track from the start it requires an agreed framework and hierarchy to help decision making and prioritization of work. Doing and implementing Agile at scale means big amount of systematic thinking and discipline where agreed approach helps to keep it on track.

In December 2019, Volvo Cars have officially completed the basic Agile transformation phase taking two and a half years, realizing that their Agile journey continues, and they are pursuing towards continuous improvements in their Agile ways of working. Nowadays, people in Volvo Cars are not anymore thinking "Agile" as a bad word, they accepted it to be a word to explain what they are doing (Denning, S. 2020).

5 IMPROVING QUALITY INTO SOFTWARE AT FASTEMS

Fastems mission is to deliver intelligent factory automation solutions with the Fastems Manufacturing Management Software (MMS) for the metalworking manufacturers in the industry which is increasingly heading towards the groundbreaking steps in digitalization. The Manufacturing Management Software (MMS) provides future-proof solutions to plan, forecast, control, visualize and monitor automated manufacturing processes (Fastems, 2019b).

Ensuring improved customer satisfaction requires also Fastems to add more effort engineering more modernized quality practices to its software solutions and products part of today's rapidly growing business environment. Customers' expectations in the factory automation industry are increasing to the level which requires a certain maturity level and turning overall software quality to be a competitive advantage. Customer satisfaction requires organizations to shift their software engineering functions from creation an engineering solution more toward satisfying the customers with their needs. Competitive products and services with high maturity level continues to be one of the key missions in the enterprises who are willing to win in the global markets and striving to gain more customers with long relationships.

5.1 Current state

Software quality control of factory automation products and solutions are evolving more from traditional testing, where quality activities takes a major part in the final stages before going to production. Fastems together with other organizations globally have had challenges to ensure high quality control for solutions of features under development as stated in the Gartner report from 2018 (Barnott, G. 2019).

The current state analyses indicated that software quality practices at Fastems have followed traditional ways of validating software. Different functions have handled their testing against own development and more integrated solution have

been tested mainly in the factory environment. Functions have had also different practices and tools for software quality control. In chapter 4.1.1 Microsoft described similar situations how their different development teams had problems to get good visibility of overall development work and see what other teams were working.

Lean, Agile, DevOps and Quality Assurance practices at Fastems has been there already some years. However, it requires to move more towards Continuous Testing method with automated testing ensuring importance of software quality control to be maintained simultaneously within software development maximizing the benefits of Agile at Scale way of working. It will create then also a better visibility and probability to act and plan faster to all required work with potential changes and additions during the Software Development LifeCycle of product or solution.

5.2 Future state creation with software quality improvements

The scale of new business demands is rapidly growing in the factory automation industry. New software solutions as well as existing product portfolio are going through a change how our industry is redefining the way how we are developing, qualifying, and delivering our software products and solutions to customers.

According to Software Engineering at Accenture “More than 50% of software development effort is spent on testing today” (Accenture. 2019). This is also coming a reality in the factory automation industry where complexity in software solutions increases all the time and requires end-to-end quality management throughout SDLC. Different roles in the organization needs to cooperate more efficiently to meet the customer satisfaction and expectations consisting objective and subjective views related to functionality, usability, and reliability.

Methods of Software Quality Engineering concept are aiming to have a rigorous check of quality in every stage of the SDLC. Furthermore, ensuring quality maintenance after the solution is delivered to customer. Increasing higher cus-

tomer satisfaction with its latest products and solutions to include extended Software Quality Assurance with Lean-Agile and DevOps practices as part of the SDLC to be continually evaluated and promoted from the very beginning. The target is now to get a Software Quality Engineering buzzword more known and in practice at Fastems.

Fastems has to prepare to a transformation become more Agile as an organization than before, empowering its organization in the future to be more collaborative, cross-functional, innovative, automated, self-managed and productive.

5.2.1 Improvements with Software Quality Engineering

A Software Quality Engineering concept will improve organization's software delivery practices to move more comprehensive E2E quality approach. It will transform organization to the mindset where quality is everyone's responsibility across organization.

A suggestion is to adapt Software Quality Engineering concept in the organization's engineering methods with revised practices of Agile at Scale, DevOps, and Software Quality Assurance. This E2E quality approach will help the Software Quality Engineering to be treated as an integrated context, improving development teams cope with various business demands and technical issues.

The suggested key actions to a Software Quality Engineering transformation at Fastems are listed below. Figure 26 together with these actions encompasses the quality-focused mindset with an expansion to enable better automated quality control of software within an Agile software delivery model.

- Ensuring to have an upgraded Quality Assurance capability in place with true Shift Left and Shift Right testing methods to combine and integrate software testing for development and operations. Increasing responsibility of everyone for Continuous Testing and enabling then Continuous Delivery.

- Renovating customer-centric software validation practices based on Test-Driven Development (TDD), Acceptance Test-Driven Development (ATDD) and/or Behavior-Driven Development (BDD) methods. These methods will build better collaboration practices and it will require every-one including the customer to keep focus in requirements and meet them properly.
- Mixing own internal quality processes to integrate Quality Engineering principles and mindset into their software development and engineering practices confirming organization's engineering outputs in high maturity level. Identify up- and re-skilling needs in the workforces with training plan to revise required capabilities and roles.
- Improving organization's engineering metrics for overall solution readiness and completion status. Increasing focus to measure predictability and vol-atility.
- Reinforcing collaboration towards cross-functional team practices and im-proving way of working with functions not operating yet with Agile practices to be stronger together. People development to support scaling-up new skills and competencies for future Software Quality Engineering needs.
- Broaden automation to plan and organize quality control for code propa-gation functions and software test validations in the most optimized way.

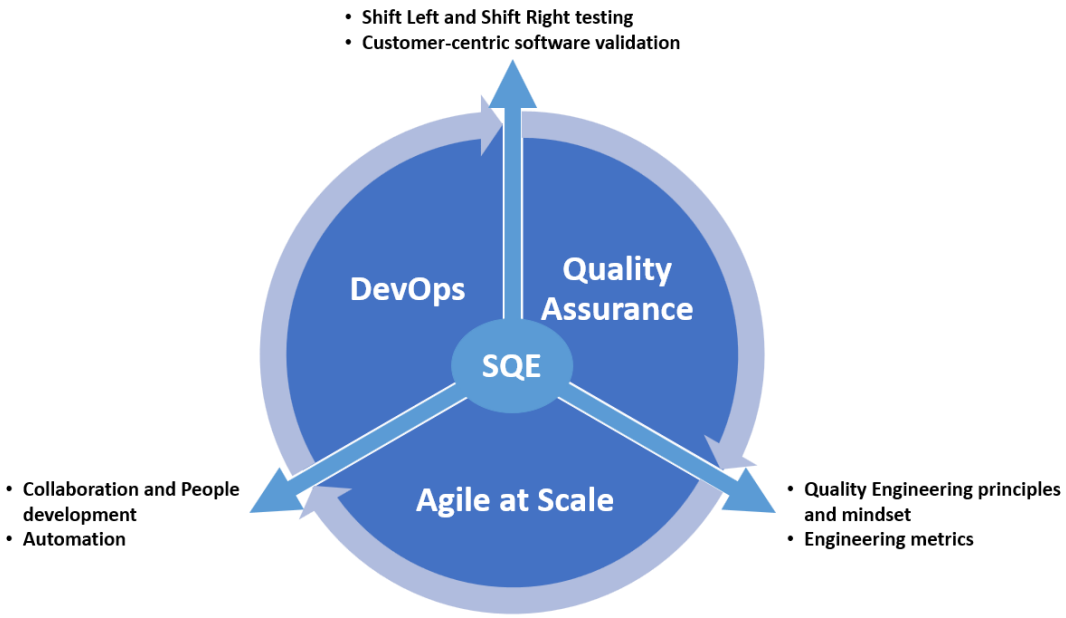


FIGURE 26. Transformation to Software Quality Engineering at Fastems.

Transforming the organization to improve overall delivery velocity, it will require from organization to adapt the new Software Quality Engineering practices, process updates and automation approaches.

5.2.2 Improvements to Software Delivery

Analysing the current state of software delivery process at Fastems addressed that validations for the major part of deliveries occurs in the final stages. This is due to fact that last couple years software solutions are including more complexity which have added more focus to build solutions properly and validate solutions more in factory environment conditions. Also, as new complex software solutions are having a tight integration to the mechanics, those has suffered from the low selection of integration testing tools.

However, the overall software delivery process can be improved in many ways and modernization steps can be divided to the different phases as illustrated in figure 27. The proposed software delivery process improvements will drive organization to the comprehensive E2E quality approach with the Software Quality Engineering described in the previous chapter 5.2.1.

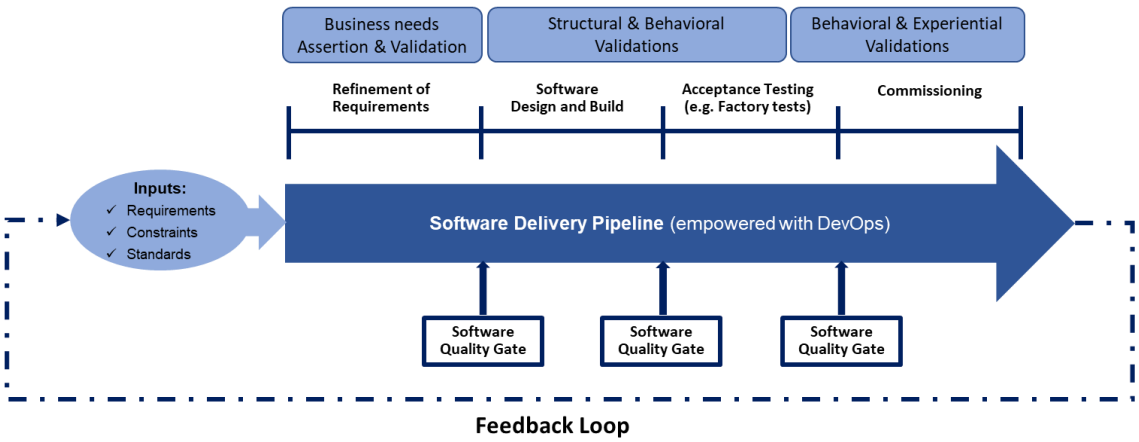


FIGURE 27. Improvements to Fastems software delivery process.

The Software Quality Engineering concept implementation into the development processes from the first stage onwards will ensure sufficient results and output, minimizing number of defects and increased costs. The principles of SQE are

based on the E2E quality management, building the mindset in organization where quality is the responsibility of everyone involved.

Each phase in the software delivery process (Figure 27) with proposed improvements has been described now in a following way:

1. **Business needs Assertion and Validation:**

The modernization of software delivery pipeline starts with business needs assertion to know well your customer with requirements, constraints and standards utilizing known collaboration model “three amigos” with the Gherkin syntax (Figure 28), omnipresent Given-When-Then style (Agile Alliance. 2020). It builds understanding to the customer expectations right from beginning between business, development including both software and hardware, as well as Quality Assurance. Getting this information right, it elevates success rate for the solution under implementation as per customer vision.

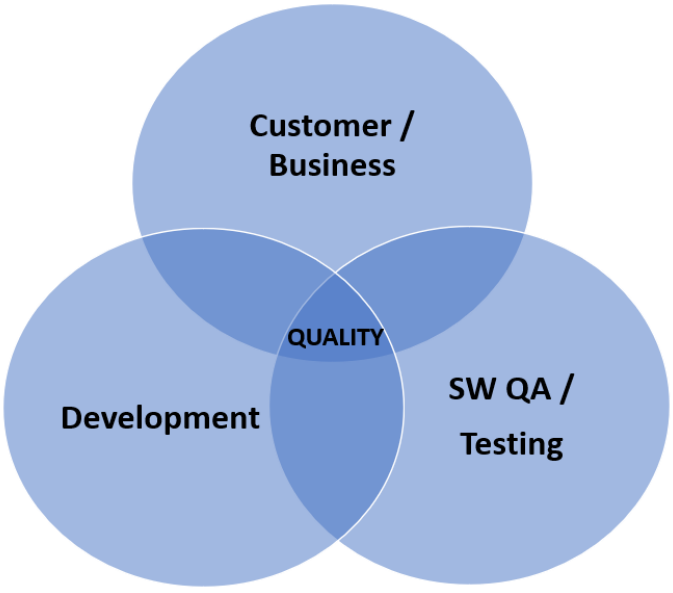


FIGURE 28. Illustration to the collaboration model of “Three Amigos”.

Short iterative process model of Agile helps to keep customer requirements up-to-date and proceeding as per expectations. Well gathered requirements will increase quality while implementing the customer solution and keeps backlog of

epics and features effective. This enables more focus to start collaboration activities as early phase as possible and continuing it all along in the software delivery pipeline to respond the feedback from different stakeholders in business and development, ensuring expected delivery with high-quality to customers. Also, overall test management planning gets more benefits with the practices to keep it effective during the complete project delivery.

2. **Structural Validations:**

Structural validations build more quality practices already before starting coding activities too heavily in the software development. Aiming to perform software unit tests with Test-Drive Development (TDD) method which is the programming practice used to create code with “test first” method, defining technical behaviour for each test. Structural validations include also analyses of code coverage and static code part of the code review practices as well as a separate quality gate checkpoint when necessary. Performance and security type of test designs and validations gets also started in this phase having then more test executions when software development is more complete.

3. **Behavioral Validations:**

Behavioral validations in the software design and build phase adds more focus then to high-level as E2E type of tests interacting with the system at granular level. Aiming to the acceptance testing activities in virtual test environments and completing them with best possible coverage in the factory testing phase. This will be done with Acceptance-Test Driven Development (ATDD) method, targeting to the widest test automation coverage against product or solution under development.

Behavioral validations with ATDD method will give indications how E2E type of tests are corresponding against customer expectations. Enabling then more visibility for all to be aware of latest findings and giving more opportunities to respond possible changes required to the software development and keeping customer up-to-date. Together they are sharing a common goal to implement and deliver

the product or solution to the customer covering their expectations with high-quality software as presented in figure 28. They are working in same understanding of customer expectations which are conversed to a customers' domain language.

4. **Experiential Validations:**

Experiential validations in the commissioning phase focus to ensure that implemented product or solution meets the customer expectations.
Experiential validations are ending to the results which can be approved by customer to start their manufacturing activities.

"Three amigos" collaboration model ensures that unexpected situations with late findings and failures are decreased to the level which can be handled efficiently and not delaying commissioning of product or solution. This model builds also high- quality of long-term documentation for the development and Quality Assurance activities. Including customer specific documentation about implemented requirements, considering required constraints and standards towards long-term relationship with agreed maintenance services.

Experiential validation phase enables important phase in the modernized software delivery pipeline to put also focus to exploratory type of testing, not only close to commissioning phase and its activities towards production, but also through the complete software delivery pipeline. Exploratory testing adds more focus to validate products and solutions with "design thinking" methods. Intention is not to document test scenarios beforehand too detail level but noting down ideas to be tested. Exploratory testing concentrates to test product or solution on the fly, simultaneously designing and executing test scenarios. It builds more freedom to discover, innovate, investigate, and learn more about product or solution.

Fastems can grow and modernize their organization notable way with Software Quality Engineering concept to be more considerable as a software company in the factory automation industry also in the future. Existing product portfolio with new software product innovations requires organization to modernize the way they develop, qualify, and deliver next products and solutions to customers.

Continuous testing method with increased automated testing processes and practices ensures to lift software quality control to the next level adding more agility, collaboration and flexibility to Software Development LifeCycle and engineering practices.

6 SCALING AGILE AT FASTEMS

Fastems's organization for software development and delivery activities has been growing rapidly during last years. New business demands includes more complexity and intelligent solutions to satisfy customers in the factory automation business. Staying competitive in the market and responding new business demands promptly requires also Fastems to continuously evaluate and how to update their Agile ways of working more towards a Lean-Agile concept. Previously commissioned Agile practices in the software development and delivery operations have built reasonable initiatives next to embrace more Agile at Scale.

6.1 Current state

This chapter will analyze the current state of Lean-Agile concept at Fastems and creates suggested improvements for its organization to embrace it more in the future. A State of Agile Survey initiated the first observations about the organization's status of Lean-Agile and how to plan improvements forward.

6.1.1 State of Agile Survey

The first State of Agile online survey at Fastems was conducted in May 2020 to evaluate the current state of Agile in the organization. A survey helped to realize how Lean-Agile concept in general is currently understood by people. It also helped to create a better view how different capabilities and competencies are connected to the Lean-Agile concept.

The first survey was focused to the organization part working in software specific functions summarizing 59 responses. The survey started with 2 clustering questions to gather relevant background information of respondents. The online survey is not including questions to collect different Agile experiences from past and compare them with the current status. This is because length of Agile experience

variates a lot between people and it is better to focus from the current status onwards.

The first background question related to the respondent's role and functional group. Respondents divided to the 3 different groups, Group A, B and C. Results presented via pie chart in figure 29 and written analyses below.

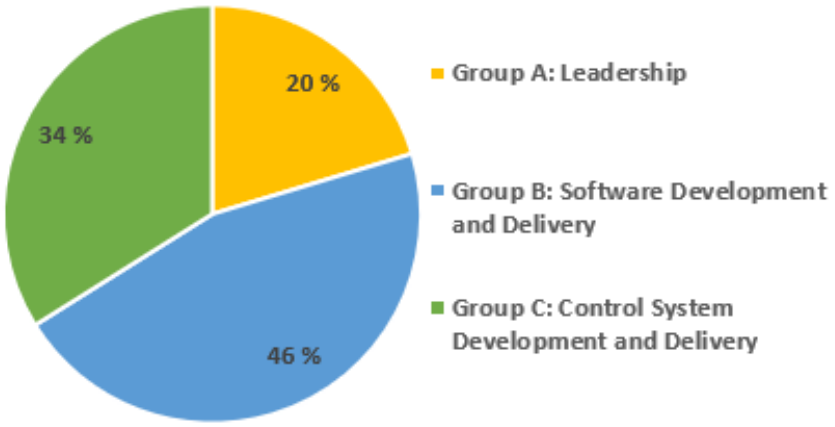


FIGURE 29. State of Agile Survey: Respondent's function and group.

- Group A represents employees in leadership positions and activities. They are people who have the power to make decisions in the line management, project, and product management. Group A received 12 responses having the 20% response rate.
- Group B represents employees working roles as a software developer, software quality engineer and software configuration engineer. Group B received 27 responses having the 46% response rate.
- Group C represents employees working roles as an automation engineer, control system engineer, commissioning engineer and training specialist. Group C received 20 responses having the 34% response rate.

The second background question related to the career years at Fastems regardless of employee's function in the organization. Figure 30 illustrates distribution of employees with their career years which indicates wide experience having majority in less than 5 career years supported by professionals with over 10 and 20 career years.

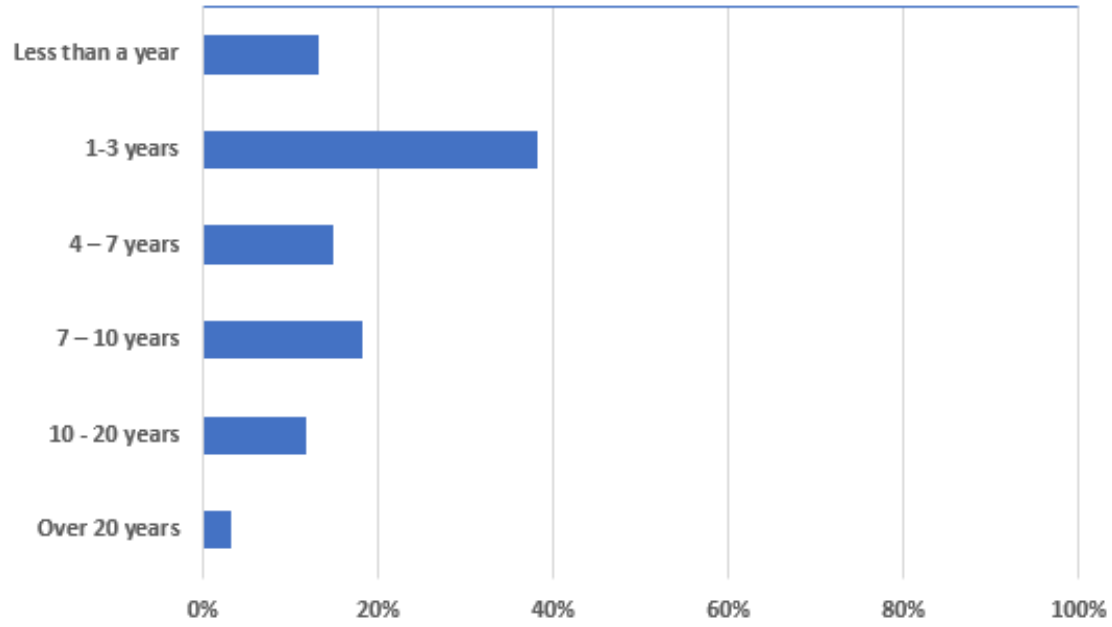


FIGURE 30. State of Agile Survey: Career years at Fastems.

The survey's actual 12 questions measuring State of Agile are representing capabilities having relation to the SAFe 4.6 Lean Enterprise Core Competencies which introduced in the theoretical background in chapter 2.4.1.

Each question starts with a short explanation of the capability. Respondents had then a short introduction to the question and request to share their estimate where we are in our Lean-Agile transformation journey of learning that capability. Survey questions includes also links to the Lean-Agile articles encouraging employees to visit sites and understand better what the capability in question is all about. Lean-Agile articles also offered to help people to learn more about concepts of Lean-Agile and SAFe 4.6 Lean Enterprise. Estimation for each question follows the scoring options 'Sit, Crawl, Walk, Run, or Fly' from the SAFe® Health Radar scale definition (Scaled Agile Framework. 2020).

The results of first online survey summarized with the radar chart to illustrate scoring results in figure 31. It shows how 12 capabilities connected to the four SAFe 4.6 Lean Enterprise core competencies: Lean-Agile Leadership, Team and Technical Agility, DevOps and Release on Demand, and Business Solutions and

Lean Systems Engineering. The radar chart illustrates coherently how scored values between capabilities have similarities or deviations.

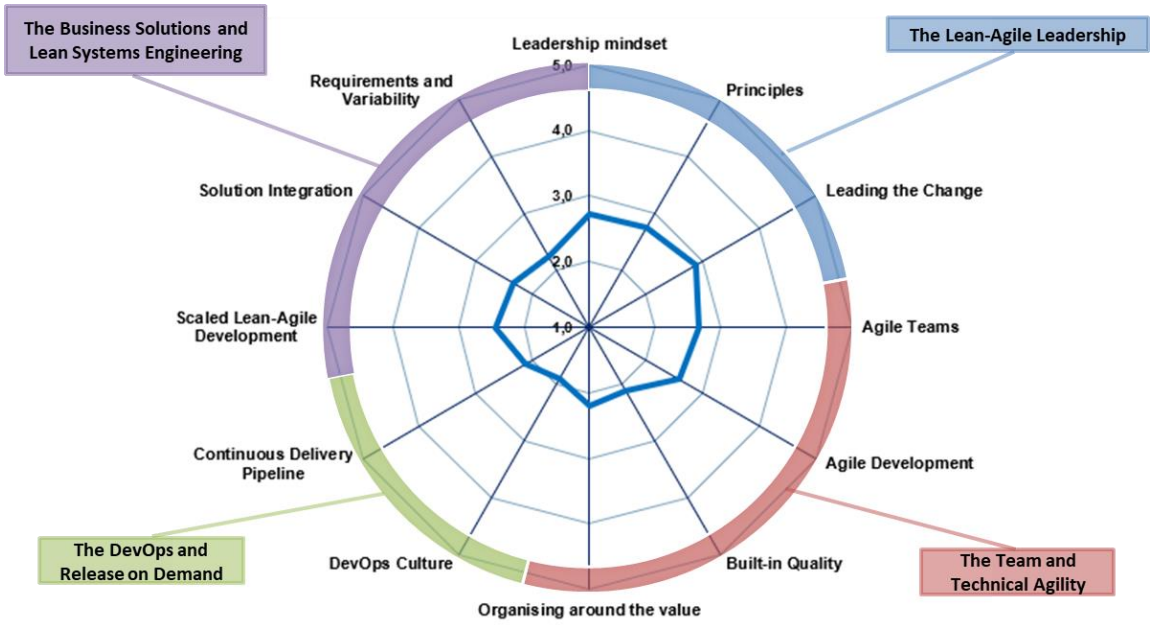


FIGURE 31. State of Agile Survey: Health Radar Assessment.

The results of first online survey are now described below per competence and its capabilities. Selected capabilities and its specific question are analyzed and illustrated via bar charts more detailed which have seen more relevant and having a best correlation for this thesis work.

1. Results for Lean-Agile Leadership

A first group of questions measured the health level of capability for Leadership mindset, Principles and Leading the Change. These three capabilities are related to the Lean-Agile Leadership competence to identify leader's ability to empower their employees through the organizational change with leading by example, changing to a new way of working and adopting a Lean-Agile mindset. Over 50% respondents estimated scoring level 'Walk' for the question 1 about Leadership Mindset and also for question 2 about Principles, meaning that leaders have been noticed to exhibit Lean-Agile mindset and behaviors in the organization.

A question 3 related to the Leading the change capability measuring how leaders are demonstrating Kotter's 8-Step Change Model part of change leadership activities (Kotter International. 2020) and how they are leading the organizational change by example. This question received close to 30% estimations for three scoring levels ‘Crawl’, ‘Walk’ and ‘Run’ as illustrated in figure 32.

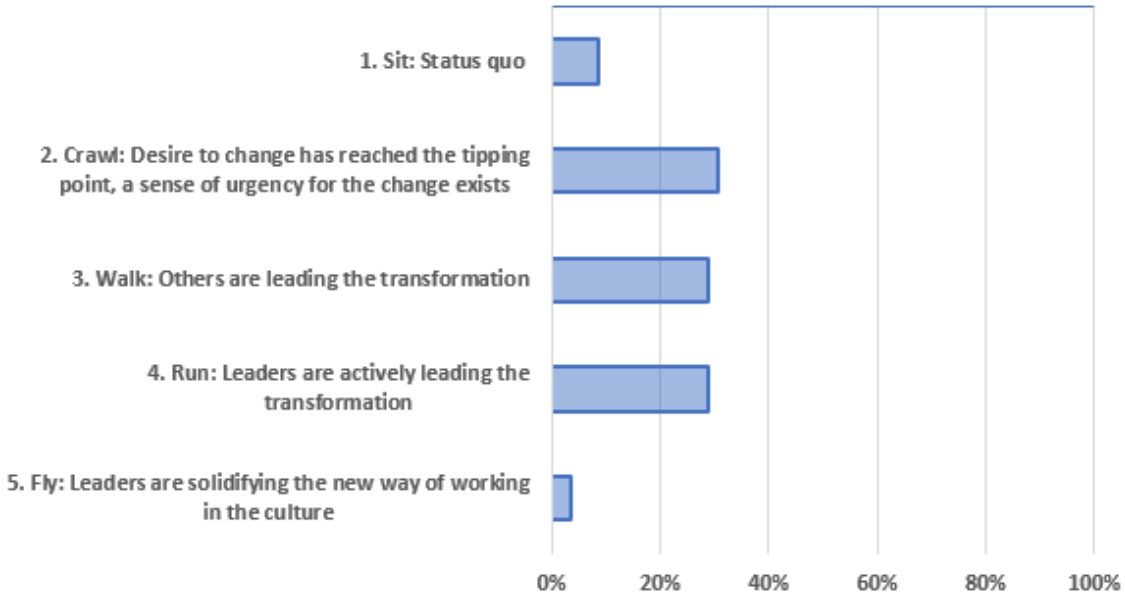


FIGURE 32. State of Agile Survey: Respondents to Leading the change (Question 3).

The result to Leading the change question indicates that majority of leaders are already taking steps leading a Lean-Agile transformation to the right direction and some even solidifying the new way of working in the culture. However, some of the leaders in ‘Sit’ needs to be checked still more detailed to ensure that they can progress towards ‘Crawl’ level. This could be related to the fact that some functions in the organization are more in the beginning of transformation than some other functions.

The positive fact for all three capabilities was that all received in average scoring level ‘Walk’ which gives a good base to continue strengthening a Lean-Agile Leadership competency via a Lean-Agile transformation.

2. Results for Team and Technical Agility

A second group of questions covers the Team and Technical Agility competence with capabilities of Agile Teams, Agile Development, Built-In Quality and Organizing around the value. This competence is targeting to measure how Agile teams are able to increase productivity, building quality in and delivering well-designed solutions with faster time-to-market.

A question 4 (Agile Teams), question 5 (Agile Development) and question 7 (Organizing around the value) assessed capabilities how team setups are currently organized around value including business and how much teams are utilizing the iterative work planning. Questions 4 and 5 received approximately 40% of respondents for two scoring levels ‘Crawl’ and ‘Walk’. Also question 7 received close to 45% of respondents for scoring level ‘Crawl’. Respondents who are estimating these capabilities be closer to ‘Crawl’ scoring level are considered to belong to teams who have recently started to take more solid actions with Agile practices. “There's still a lot of work to do even we have taken steps to right direction”, wrote one of the respondents.

A question 6 results examined via figure 33. This question targeted to measure how much development still relies to traditional testing strategies and also how test automation together with ‘Shift-Left’ testing strategies are currently utilized.

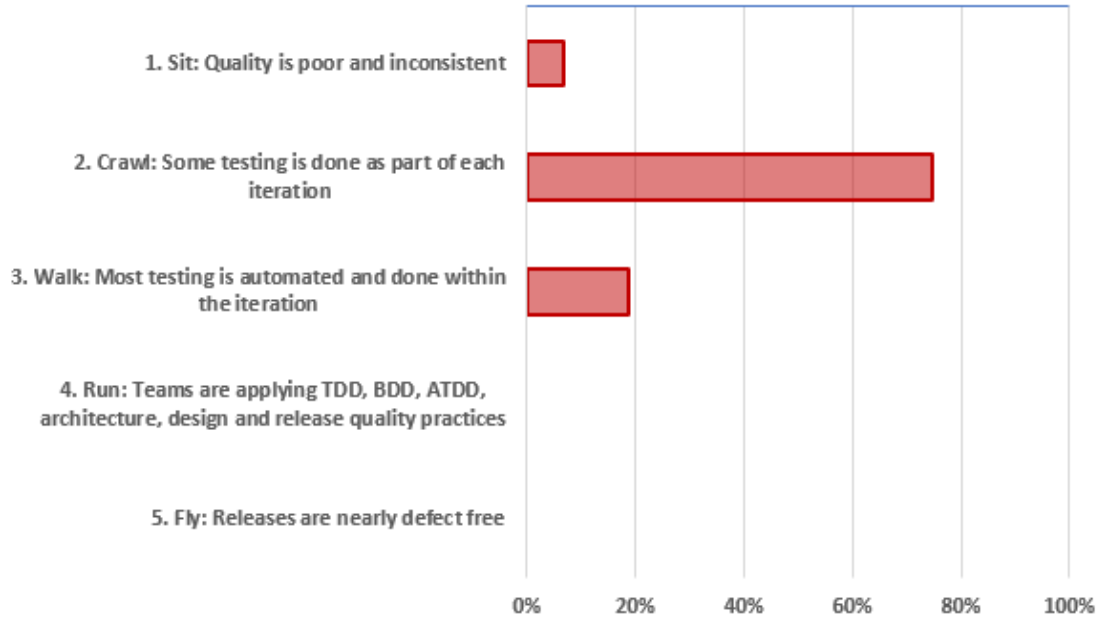


FIGURE 33. State of Agile Survey: Respondents to Built-In Quality (Question 6).

The results indicates that traditional testing methods has been mainly used even test automation is now more and more strongly coming part of daily development activities throughout various technologies in use.

Respondents shared also comments that some technologies can be test automated and also taking ‘Shift-Left’ testing strategies in use must faster than others due to more comprehensive testing tool options. Especially technologies suffers options to develop testing more to automated solutions which are dependent on about mechanics and hardware engineering. This have led to the situations where major part of testing is completed in final stages of project.

The capabilities in this Team and Technical Agility competence received in average scoring level ‘Crawl’ which indicates that ongoing improvements in the organization are not visible yet so well. Positive thing is that actions around this competence are in good progress and estimating to move towards ‘Walk’ and ‘Run’ scoring levels already in next survey rounds. Also based on the feedback from respondents this question requires some fine-tuning to describe levels better to fit for factory automation industry.

3. Results for DevOps and Release on Demand

A third group of questions measured the health level of The DevOps Culture and Continuous Delivery Pipeline capabilities for the DevOps and Release on Demand competence. These questions measures organization capability for better collaboration to align customer needs and releasing value to customers more promptly.

Feedback from respondents for the question 8 about the DevOps Culture separates respondents views how this have seen part of the current way of working. Silos between functions have seen still one of main issues, but latest organization updates are now removing remaining silos and improving the collaboration. However, respondents also commented that DevOps Culture has been part of working culture in the organization already very long time. During survey analyses this question agreed to be taken under further development and define it better to support Fastems DevOps Culture and its actions to move to the next levels.

A question 9 in figure 34 assessed the Continuous Delivery Pipeline capability which includes four parts: The Continuous Exploration, Continuous Integration, Continuous Deployment and Release on Demand. Estimates shows that respondents have seen already the Continuous Integration and Continuous Deployment to be covered in the current development activities even a weight is more still in 'Crawl' level with 61% result. Also 'Walk' level received 20% result, so some teams already demonstrating the largely automated Continuous Integration and Continuous Deployment capabilities part of their project deliveries.

DevOps and Release on Demand competence with its capabilities assessed to have similar scoring level 'Crawl' in average than in the previous competence. Based on the respondent's feedback for the capabilities in this group requires better level definitions how this will be estimated in the next survey rounds. Also, more dialog with teams has been requested to set targets having more company-specific to correspond better to requirements in the factory automation industry.

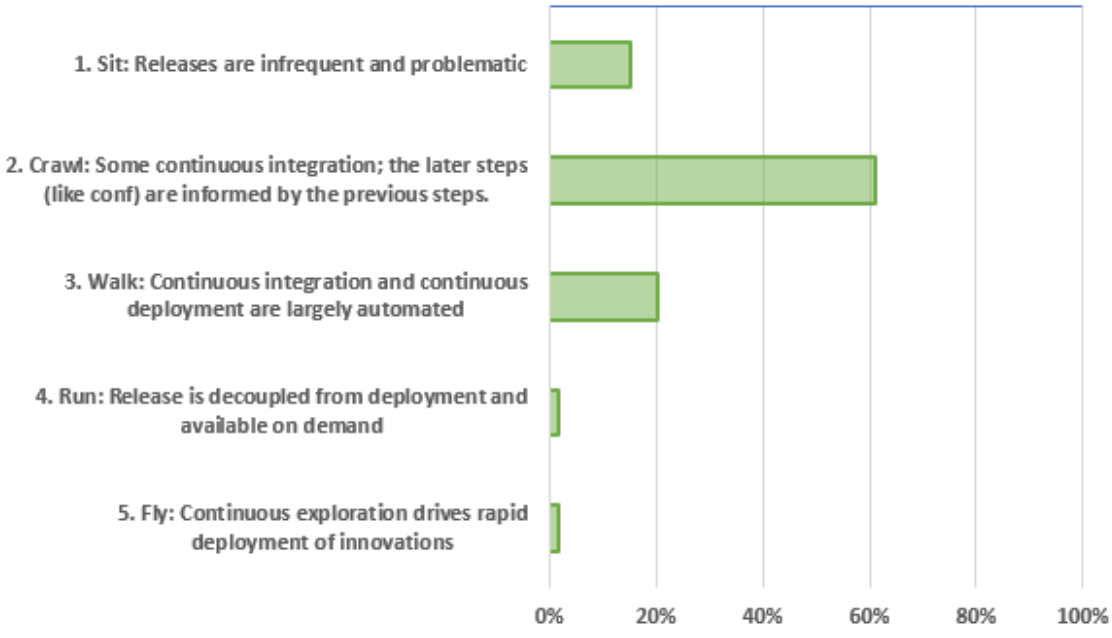


FIGURE 34. State of Agile Survey: Respondents to Continuous Delivery Pipeline (Question 9).

4. Results for Business Solutions and Lean Systems Engineering

A fourth and final group of questions were for the Business Solutions and Lean Systems Engineering competence. This group of questions includes capabilities for Scaled Lean-Agile Development, Solution Integration, and Requirements and Variability. This competence measures organization capability to scale its activities with provided tools, ability to produce customer solutions faster and enabling better predictability.

A question 10 in figure 35 illustrates how the current level of scaled agile development is estimated at Fastems. There is only so much that an Agile team can accomplish. When projects are getting larger and more complex than those are coming too big to handle by one Agile team. It requires the organization to scale its development functions and doing them in a larger synchronized set. More than half of respondents (53%) estimated that Teams of Agile Teams are somehow formalized and coordinated in the organization already.

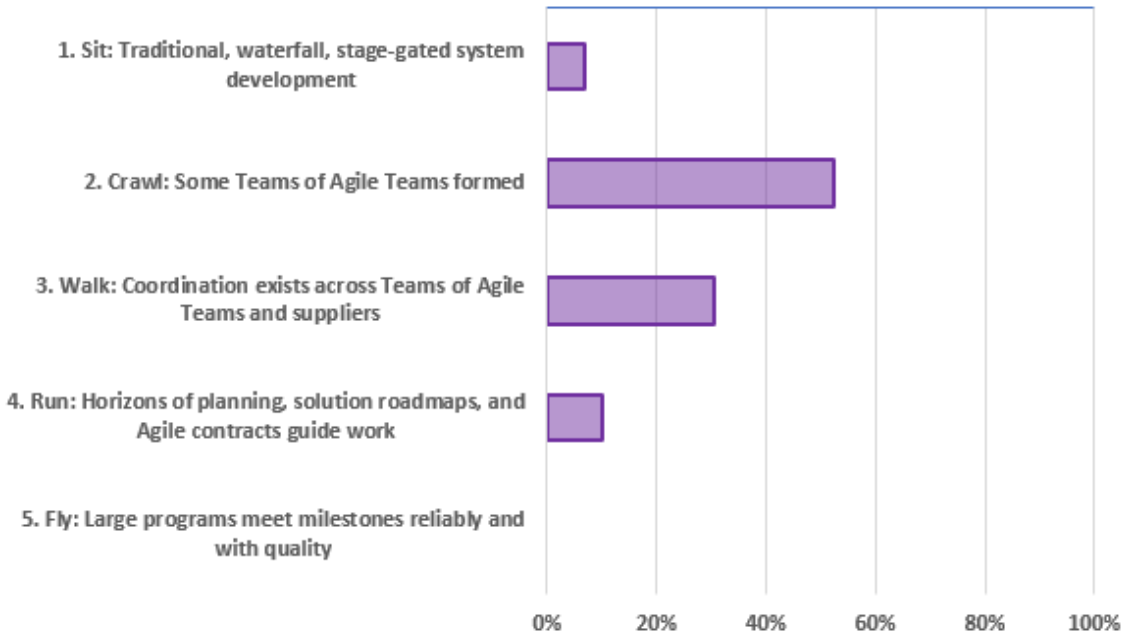


FIGURE 35. State of Agile Survey: Respondents to Scaled Lean-Agile Development (Question 10).

However, in a question 10 a weight stayed closer in ‘Crawl’ level in this first online survey. Recent actions will give high probability to get a more weight towards ‘Walk’ and ‘Run’ levels with increased awareness of Agile team coordination and planning activities.

A question 11 focused to the Solution Integration capability to measure the level of combining more demanding blocks together. When starting to deal with either larger than normal software solutions or with solutions that are referred to be as cyber-physical solutions (includes both hardware and software), then organization will face new challenges in Continuous Integration. This question estimated with 53% respondents to be in ‘Crawl’ level seeing frequent integration of software elements and 31% respondents estimated it to ‘Walk’ considering having elements in place for regular full solution integration.

A question 12 closed the survey with Requirements and Variability capability assessment to measure how organization is exploiting variability in the requirement management. The organization have to deal with more complex solutions and customer needs. Therefore, requirements need to be handled differently. Respondents with 70% estimate are seeing some requirements and design having variability in place. It states level now to ‘Crawl’ in the first online survey. The rest of respondents (25%) estimated level already to ‘Walk’ or even ‘Run’. However, this was also one of the questions agreed to be taken under further development and define it better to support organization needs and targets for the future.

A table 1 illustrates gathered general feedback from respondents for 4 capabilities (Leading the Change, Built-in Quality, Continuous Delivery Pipeline and Scaled Lean-Agile Development) which were analyzed at this first State of Agile online survey. All estimated results and feedback from this survey will help organization to take right steps forward and concentrating to make it in a way that helps adapting to Lean-Agile capabilities better.

TABLE 1. State of Agile Survey: Gathered general feedback.

SAFe Lean Enterprise core competency	Lean-Agile Capability	General feedback
The Lean-Agile Leadership	Leading the Change	▪ "…Generally speaking, the will towards more textbook lean-agile seems to mostly exist at all levels and is not a huge problem. Problems arise with practicalities of the nature of our business, historical ways of working and lack of time to focus on change (which always initially increases workload)…" ▪ "…to succeed in the agile transformation, it is important to not only focus on certain teams…"
The Team and Technical Agility	Built-in Quality	▪ "…only see the end results with what is delivered and what the customer expects…" ▪ "…the requirements aren't so clear…"
DevOps and Release on Demand	Continuous Delivery Pipeline	▪ "…the continuous releases don't really work that well with one-off projects. Of course, in longer projects you can have multiple releases for customer demos etc…"
Business Solutions and Lean Systems Engineering	Scaled Lean-Agile Development	▪ "…on site during commissioning, the agile teams work very well in Fastems currently. Development before commissioning is more siloed…"
General feedback	About questions and survey in general	▪ "Questions were too complex. Next time more simple ones…" ▪ "Questions had too much jibberjabber and main point was sometimes difficult to point out…" ▪ "Many of the question topics were completely new to me." ▪ "Some of the answers depend a lot of which department we are talking about." ▪ "In many cases, achieving Fly or Run state is very difficult or even impossible. For example, continuous delivery is aspect which seems very hard to achieve."

The feedback from respondents illustrates well how important view the general feedback always creates. It tells honestly how respondents have understood the content of survey and what is good improve in the next survey round. It helps people who are maintaining a survey content to update it to the level which helps respondents better to understand all questions and material share through a survey.

6.1 Future state creation with Lean-Agile improvements

Fastems's journey to move more towards Lean-Agile concept requires to concentrate building a sustainable path with new Agile framework. Offering of Agile frameworks which supports Lean-Agile concept varies with different models how to scale Agile across the organization and any model itself is not the whole truth.

It will require leaders in the organization to evaluate selected model and its practices continuously, shifting organization to act as a learning organization. A learning organization means that during a transformation leaders and teams are getting relevant trainings to understand principles better in their everyday work and also having coaching support available to understand different Lean-Agile behaviors at all organization levels.

6.1.1 Improvements via Lean-Agile transformation program

Enterprises worldwide are strongly believing that SAFe framework gives for them to give best capabilities to scale Agile in their organizations. Chapter 4.2 described two stories how Volvo Cars and John Deere have been able to move their organizations to use SAFe framework. Same chapter reported the results from the KPMG's Survey on Agility (KPMG. 2019) and also from the Gartner's enterprise Agile survey (Gartner.2019). Both survey results concluded that enterprises worldwide are highly considered to proceed their next Agile transformation with Scaled Agile Framework (SAFe®) successfully integrate their business and IT functions.

This is also suggested at Fastems to proceed with a new Lean-Agile transformation program. A Lean-Agile transformation program is targeting to use a systematic approach to find and lead a well-disciplined path for Fastems to scale its Lean-Agile implementation. It will establish a new ways of working to develop both software and hardware simultaneously in an integrated fashion.

Suggested SAFe framework gives an opportunity to continuously evaluate its ability to refresh Fastems's knowledge base for Lean, Agile, and DevOps with

SAFe principles, practices, and competencies. It gives support with guidelines to refine roles and responsibilities in a way to support organizational growth and activities to continue staying competitive against other competitors. It provides tools and methods to align, collaborate and deliver more efficiently with multiple Agile teams across the organization, improving productivity, quality, and employee engagements.

Providing SAFe trainings for people to understand what SAFe really is and how it could work in practice. SAFe framework ensures also that implementation of Agile at Scale goes to right track from the very beginning and helps people in decision making and prioritization of work. Agile at scale under practice and implementation means big amount of systematic thinking and discipline where agreed approach helps to keep it on track.

Also already concluded State of Agile Survey supports a Lean-Agile transformation with introduced SAFe core competencies and its capabilities. A survey can be considered to perform in frequent intervals to follow the progress how organization adapts to the new concept. It also helps to identify better the next steps to be improved in a Lean-Agile transformation journey.

6.1.2 Improvements to Lean-Agile change leadership

A Lean-Agile transformation program requires a strong leadership skill from the leaders in the organization to lead large group of people be modernized and transformed to the new Lean-Agile way of working and empowering them for the change.

The leaders in the organization needs to understand and learn more deeply how their workforces are currently actually delivering the products and solutions. While organization is growing rapidly, evaluation of the current delivery practices are building also more trust for people and increasing better visibility across the entire organization and product delivery lines. Like said in chapter 2.6.1, at the end of the day, transformations is about the people and how they work. That is

why the leaders in the organization must prepare well how they treat their people through the transformation efficiently and successfully.

A Lean-Agile transformation program is targeting to take more steps towards human-based approach to appeal better to people feelings, having more focus in the people who will implement the vision of change and bringing it to reality.

The visibility of the progress in a Lean-Agile transformation program is ensured from the very beginning, to give everyone to follow it readily. A Lean-Agile transformation program ensures that a guiding coalition exists and continuously sharing latest information and feeding quick wins to all parties early and often. The transformation program includes reasonable amount of people who could resist the change and collectively sharing targets for the people, driving them to the same direction. Previous processes or practices to be evaluated, removing, or updating them to support execution of transformation more efficiently.

A psychological trust will play also important part in the transformation to build an environment for the people where they can share their best knowledge, thoughts, feelings, and views without fare of any negative consequences. This will create trusted collaboration between teams and people in transformation to allow openness and trust-worthiness.

As most of the digital transformations also this one at Fastems will arise various feelings in people like curiosity, passion, confusion or even irritation. These must be taken well aware and support them during the transformation, giving then more space for people to innovate and set their goals to support transformation comfortably to the success. The transformation requires that it will be taken as a journey which never ends and is continuously evaluated with surveys, streamlined to support latest organizational changes and business updates.

7 RESULTS

Fastems, together with enterprises globally are highly considered proceeding digital transformation with investments scaling agility and quality engineering in their organizations in the next three years. Key drivers are related to modernize the software delivery operations with Software Quality Engineering concept to enable more speed to product and solution deliveries with continuously improved customer satisfaction. In addition, to increase more agility across organization with new Lean-Agile way of working between business and IT functions.

As analysed case studies in chapter 4 enterprises like Microsoft and Google have used a great effort with their digital transformations. These industrial revolutions have engaged their teams become more effective and operating faster with better quality. Fastems will have a same target with own transformation journeys in the new future.

7.1 Steps forward with Software Quality Engineering

Fastems opportunity to proceed with concept of Software Quality Engineering within Lean-Agile requires their organization shifting roles, processes, and tools to optimize its SDLC with automation-driven approaches. These improvement suggestions will help organization to adapt to the comprehensive E2E quality approach using Software Quality Engineering.

Each case study in chapter 4 revealed that each company realized that quality is everyone’s responsibility in the organization. They are involving own organizations to be a part of decision making throughout SDLC to manage quality control. However, this requires a great effort from the organization to shift their forces to follow these targets and continuously evaluate the progress.

Suggested future state improvement plan for software quality at Fastems described in chapter 5.2. This transformation journey will also take a great effort

from Fastems and needs to have clearly defined actions forward. Table 2 summarizes first key actions with revised practices of Agile at Scale, DevOps, and Software Quality Assurance.

TABLE 2. First key actions towards Software Quality Engineering at Fastems.

Revised SQE practices	Improvement targets	Improvement suggestions
Agile at Scale	▪ Collaboration and People development	▪ Refining the roles with combination of new roles and renovated responsibilities. ▪ Increasing teams and individual's technical proficiency across the organization.
DevOps	▪ Quality Engineering principles and mindset ▪ Engineering metrics	▪ Clear quality control guidelines for tools, data, and environment management to support automation. ▪ DevOps integration and test execution infrastructure.
Software Quality Assurance	▪ Shift Left and Shift Right testing ▪ More automation ▪ Customer-centric software validation	▪ Practicing more towards In-Sprint and Cross-Sprint deliveries and enabling more automation-driven approaches part of them. ▪ Having common language and definition for the different type of test activities including customer interface.

Transforming the organization to the Software Quality Engineering concept it improves overall delivery velocity. Those revised practices of Agile at Scale, DevOps, and Software Quality Assurance are optimizing organization's SDLC with automation-driven approaches.

7.2 Steps forward with Lean-Agile

Chapter 6.1 created the baseline to plan future state for Lean-Agile practices at Fastems. It proposes to build a Lean-Agile transformation with 12 capabilities which have a solid relation to the SAFe Lean Enterprise core competencies: Lean-Agile Leadership, Team and Technical Agility, DevOps and Release on Demand, and Business Solutions and Lean Systems Engineering.

Case studies for Volvo Cars and John Deere in chapter 4 clarified that Agile at Scale for every organization is an extensive cultural transformation to commit

workforces from different functions. It aims to harmonize practices and tools for efficient collaboration, decision-making and transparency. Global enterprises like Volvo Cars and John Deere are aware that to stay competitive in their business requires them to be closer to customer with their needs and creating more flexibility in their offerings. When organization initiates this kind of cultural transformation, it requires to be taken as a journey which never ends.

The first State of Agile survey at Fastems was also a one good lesson learnt while considering establishing a Lean-Agile transformation that any model itself is very hard to be adapted directly to the own organization needs. It will require enterprise to act as a learning company to evaluate selected model and practices continuously. Table 3 clarifies the main improvements to be implemented in the organization once proceeding forward with a Lean-Agile transformation.

TABLE 3. Improvement actions establishing Lean-Agile transformation at Fastems.

Lean-Agile topics	Topic description	Improvement suggestions
Lean-Agile transformation program	▪ To use a systematic approach to find and lead a well-disciplined path to practice a Lean-Agile concept at Fastems	▪ Establishing a Lean-Agile Center of Excellence ▪ Creating a backlog of improvements which will be implemented as an own separate development projects.
Lean-Agile capabilities	▪ Introduced 12 Lean-Agile capabilities are relatively new topic still for most of the people.	▪ More communication and training in the organization to understand them better and how these have relation to own teamwork. ▪ Each capability needs to be practiced part of own Agile team a bit longer and understood still better that how it will connect to the organizational culture of Lean-Agile.
State of Agile survey	▪ How ranking is done for each capability using the SAFe® Health Radar scale	▪ The SAFe® Health Radar scale has to be trained and explained better for the people that how it works and where it aims for. ▪ A survey questions requires to be checked through and modified closer to the factory automation industry and Fastems business environment.

These first improvement actions would lead a Lean-Agile transformation journey to the right direction to be a success in the future. A new program will set a clear centre point to lead, implement, and communicate all Lean-Agile improvements

to the organization. Lean-Agile capabilities requires more practicing and training to embrace better part of own daily work activities. The survey data has to be more accurate in the next survey rounds. Giving more accurate results how teams are reflecting true health and maturity against each capability which we are learning and practicing part of Lean-Agile transformation journey.

Fastems change leadership is preparing for a new phase. Leaders at Fastems are familiarizing themselves what kind of benefits scaling up Agile practices throughout the organization can provide. They want to understand also better steps to be taken to lead transformation more reasonable and realistic ways which fits best for own organization setup and business. Ensuring successful steps forward in a Lean-Agile transformation requires a strong enterprise leadership commitment, avoiding model where transformation is tried to be managed and handled bottom-up.

A one key step next is that leaders at Fastems have to prepare for an evaluation with different functions in the organization how well Lean-Agile practices can be organized to the team or can it be implemented to the team at all. Teams who cannot operate with Lean-Agile practices have then right tools and model to support teams who are doing Lean-Agile. The article in chapter 2.5.1 from Talouselämä magazine (Talouselämä. 2018) supports this approach to be followed. The article reported that enterprises should move their transformations towards the human-based approach. Adding more focus to people feelings and collaboration with people who implementing the vision of change and bringing it to reality.

8 DISCUSSION

The purpose of this thesis was to provide an overview how Fastems together with enterprises worldwide are modernizing and engineering quality into their software delivery operations. The industrial revolution is ongoing in digitalization and used software technologies which causing organizations to refine their working cultures continuously. Enterprises are adding more focus to satisfy their customers engineering solutions based on needs and where a quality plays a major role. A quality has become a critical attribute to ensure a best user experience with long relationships with customers.

Case studies summarizes how enterprises worldwide are taken in use Software Quality Engineering concept and scaling their Agile practices throughout organizations. However, case studies also described how enterprises can face obstacles or even fail these transformation programs with unprofessional change leadership. This adds a clear need for enterprises continuously evaluate what they are aiming for with ongoing transformations. They have to be ready to update or change transformation targets to ensure successful journey and business impact.

Fastems continues in a good position in the factory automation business as it invests to the future to modernize their software delivery operations. Recent updates to support Software Quality Engineering concept are already clearly visible. Fastems organization is started to grow their software delivery workforces with right knowledge and experience. New investments for better quality control are under way and near future will show how many described improvements can be implemented already be end of this year.

Software Quality Engineering concept and suggested improvements for Fastems how it was described in this thesis, could take solid steps forward engineering higher quality into their software delivery operations. Fastems is moving more to comprehensive E2E approach in quality management and ownership mindset, where quality is become everyone’s responsibility across organization.

Fastems is taking also steps forward with Lean-Agile transformation and initiating a new program which clearly remarking to keep its organization competitive in today's rapidly growing business environment with the ever-increasing scale of new business demands.

A State of Agile survey addressed already that Fastems with its organization is ready to steps forward and adapt to the Lean-Agile transformation and taking it as a journey which never ends. It will add organization to act in a continuous learning mode to evaluate selected concepts and practices every now and then.

In the future, State of Agile survey brings for the organization added value with its quantitative data. It increases understanding how are we proceeding, where are we on this journey at the moment and which are the next steps to improve the journey. It is already considered to be performed in frequent intervals to follow the progress and able to identify better the next steps to be improved in a Lean-Agile transformation journey.

Communication of State of Agile survey results are coming a common part of organization's monthly meetings to follow the progress of Lean-Agile transformation journey. The survey brings one important interaction channel for the people. It adds more confidence around shared information and overall communication about ongoing activities. Also, decision making improves when it will be based on data collected together. People feels that they are listened and getting more support when needed. It will build better commitment with people and increases value to the company and our customers.

REFERENCES

Accenture. 2019. Article. Leading quality engineering into the New. Read 27.12.2019. https://www.accenture.com/gb-en/insights/technology/quality-engineering-new?src=SOMS

Agile Alliance. 2020. Acceptance Test Driven Development (ATDD). Read 2.2.2020. https://www.agilealliance.org/glossary/atdd/

Atlassian. 2019. Article. Agile at scale. Read 2.2.2020. https://www.atlassian.com/agile/agile-at-scale.

Avram, A. 2011. Article. Ensuring Product Quality at Google. Read 11.1.2020. https://www.infoq.com/news/2011/03/Ensuring-Product-Quality-Google).

Baltierra, C. 2016. Article. Read 11.1.2020. Is Quality Engineering a Dying Art? https://engineering.salesforce.com/is-quality-engineering-a-dying-art-d75fab1963d9).

Barnott, G. 2019. Article. 5 ways Quality Engineering can streamline your digital transformation. Read 19.1.2020. https://www.softwaretestingnews.co.uk/5-ways-quality-engineering-can-streamline-your-digital-transformation/

Bird, J. 2013. Article. This is how Facebook develops and deploys software. Should you care? Read 11.1.2020. https://dzone.com/articles/how-facebook-develops-and).

Blazemeter. 2019. Article. What is Shift-Right Testing? Read 2.2.2020. https://www.blazemeter.com/blog/what-is-shift-right-testing/.

Bloomberg, J. 2014. Article. Scaling Agile Software Development for Digital Transformation. Read 8.2.2020. https://www.forbes.com/sites/jasonbloomberg/2014/09/08/scaling-agile-software-development-for-digital-transformation/#298bcb00438d

Bourque, P., R. Dupuis, A. Abran, J.W. Moore, L.L. Tripp et S. Wolff, (2000) Fundamental Principles of Software Engineering - A Journey, Journal of Systems and Software.

Breu R, Kuntzmann-Combelles A, Felderer M. 2014. New perspectives on software quality. IEEE Software. Read 12.1.2020. https://ieeexplore-ieee-org.libproxy.tuni.fi/document/6750444.

Bright, P. 2014. Article. How Microsoft dragged its development practices into the 21st century. Read 28.12.2019. https://arstechnica.com/information-technology/2014/08/how-microsoft-dragged-its-development-practices-into-the-21st-century/4/.

Cardoza, C. 2017. Report. Agile and DevOps provide more benefits together than alone. Read 9.2.2020. https://sdtimes.com/agile/report-agile-devops-provide-benefits-together-alone/

Côté, M-A. & Suryn, W. & Georgiadou, E. Article. Software Quality Model Requirements for Software Quality Engineering. Read 27.12.2019. https://www.researchgate.net/publication/228947488.

Denning, S. 2015. Article. Surprise: Microsoft Is Agile. Read 27.12.2019. https://www.forbes.com/sites/stevedenning/2015/10/27/surprise-microsoft-is-agile/#1192eb622867.

Denning, S. 2020. Article. Why And How Volvo Embraces Agile At Scale. Read 24.2.2020. https://www.forbes.com/sites/stevedenning/2020/01/26/how-volvo-embraces-agile-at-scale/#2c8c7e504cf0.

Dotterweich, A. 2018. Article. What Growing Teams Can Learn about QA from How Google Tests Software. Read 11.1.2020. https://www.rainforestqa.com/blog/2018-01-11-what-growing-teams-can-learn-about-qa-from-how-google-tests-software/.

Fastems. 2019a. Company Description. Read 27.12.2019. https://www.fastems.com/company.

Fastems. 2019b. Fastems MMS Brochure. Read 27.12.2019. https://www.fastems.com/offering/mms/.

FusionCharts. 2020. Radar (Spider) Chart. Read 12.4.2020. https://www.fusioncharts.com/resources/chart-primers/radar-chart

Galin, Daniel. 2004. Software Quality Assurance: From theory to implementation. Pearson Education Limited, England.

Gartner. 2019. Survey Analysis. Enterprise Agile Frameworks Maximize Potential for Achieving Agility at Scale. Read 8.2.2020. https://www.gartner.com/doc/reprints?id=1-1OB0FGQN&ct=190723&st=sb.

Herschmann, J. & Murphy, T & Scheibmeir, J. 2019. Magic Quadrant for Software Test Automation. Gartner.

InfoQ. 2012. Article. Your Tractor Was Built With Agile. Read 22.3.2020. https://www.infoq.com/news/2012/03/deere/

Kotter International. 2020. Company Brochure. 8 steps to accelerate change in your organization. Read 25.2.2020. https://www.kotterinc.com/.

Kotter, J. 2011. Article. Change Management vs. Change Leadership - What's the Difference? Read 25.2.2020. https://www.forbes.com/sites/johnkotter/2011/07/12/change-management-vs-change-leadership-whats-the-difference/#

KPMG. 2019. Survey on Agility. Agile Transformation: From Agile experiments to operating model transformation: How do you compare to others? Read 25.2.2020. https://assets.kpmg/content/dam/kpmg/nl/pdf/2019/advisory/agile-transformation.pdf

McKinsey & Company. 2016. Article. The ‘how’ of transformation. Read 25.2.2020. https://www.mckinsey.com/industries/retail/our-insights/the-how-of-transformation.

Meier, J. 2014. Article. Quality Engineers and Developers… Can’t we all just get along? Read 11.1.2020. https://developer.salesforce.com/blogs/engineering/2014/05/quality_engineers_and_developers_cant_we_all_just_get_along.html).

Mistrik, I. & Soley, R. & Ali, N. & Grundy, J. & Tekinerdogan, B. 2016. Software Quality Assurance: In Large Scale and Complex Software-Intensive Systems. Morgan Kaufmann Publishers.

Pugh, Kenneth. 2011. Lean-agile acceptance test driven development: better software through collaboration. Addison Wesley Professional. Upper Saddle River, NJ.

Quality Engineering. A diagram of quality engineering – E2E software quality. Read 11.1.2020. https://www.wikiwand.com/en/Quality_engineering#/overview).

Razzak, M. & Richardson, I. & Noll, J. & Canna, C. & Beecham, S. 2018. Scaling agile across the global organization: an early stage industrial SAFe self-assessment. Read 8.2.2020. https://andor.tuni.fi/permalink/358FIN_TAMPO/1msm64c/acm3196373

Rigby, D.K., Sutherland, J., Noble, A. 2018. Article. Agile at Scale. Read 2.2.2020. https://andor.tuni.fi/permalink/358FIN_TAMPO/1msm64c/scopus2-s2.0-85055039057

Software Testing Help. 2019. Top Software Testing Trends To Follow in 2020. Read 11.1.2020. https://www.softwaretestinghelp.com/software-testing-trends/.

Suryn, W. 2014. Software Quality Engineering: A Practitioner's Approach. John Wiley & Sons, Inc., Hoboken, New Jersey.

Scaled Agile. 2020. Guiding the Lean Enterprise. Read 8.2.2020. https://www.scaledagile.com/

Scaled Agile Framework. 2020. Article. Case study: John Deere. Read 8.2.2020. https://www.scaledagileframework.com/john-deere-case-study/

Talouselämä. 2018. Article. Mielipide: Muutosjohtajan on valjastettava tunteet - ja näin se tehdään. Read 5.5.2019. https://www.talouselama.fi/uutiset/mielipide-muutosjohtajan-on-valjastettava-tunteet-ja-nain-se-tehdaan/0c28dfa1-d7f4-3f8b-86e7-ed80c20a43f6

Thomas, N. 2018. Article. A Definition of DevOps for the Masses. Read 9.2.2020. https://devops.com/definition-devops-masses/.

Thomson, R. 2008. Lack of software testing to blame for Terminal 5 fiasco. Article. Read 6.1.2020.

http://www.computerweekly.com/Articles/2008/05/08/230602/lack-of-software-testing-to-blame-for-terminal-5-fiasco-ba-executive-tells.htm

Tian, J. 2005. Software Quality Engineering: Testing, Quality Assurance, and Quantifiable Improvement. John Wiley & Sons, Inc., Hoboken, New Jersey.

Tozzi, Christopher. 2018. Article. How DevOps Teams Can 'Shift Left' and Shift Right' Their Way to an Effective Continuous Development Model. Read 2.2.2020. https://andor.tuni.fi/permalink/358FIN_TAMPO/1msm64c/proquest2056794042

Tricentis. 2018. Article. Why (and How) Manual Testing Must Change. Read 19.1.2020. https://www.tricentis.com/blog/why-how-manual-testing-must-change/.

Van Brummelen, J. & Slenders, T. 2019. Article. Modern Software Development: It is all about quality and speed. Read 20.1.2020. https://www.compact.nl/en/articles/modern-software-development/

Verona, J. 2016. Practical DevOps: harness the power of DevOps to boost your skill set and make your IT organization perform better. First edition. Packt Publishing Ltd. UK. ISBN 978-1-78588-287-6

Walsh, Tanja. 2016. Online Surveys in Collecting Cross-Cultural Qualitative User Experience Feedback. Read 13.4.2020. http://urn.fi/URN:ISBN:978-952-15-3733-2.

Wiklund, K. & Eldh, S. & Sundmark, D. & Lundqvist, K. 2017. Software Testing, Verification and Reliability. Read 12.01.2020. https://onlinelibrary-wiley-com.libproxy.tuni.fi/doi/full/10.1002/stvr.1639.

Wikipedia. 2020. Rogers Technology Curve. Read 24.2.2020. https://en.wikipedia.org/wiki/Technology_adoption_life_cycle.

Zobel, S. 2018. Why Digital Transformations Fail: Closing The $900 Billion Hole In Enterprise Strategy. Read 25.2.2020. https://www.forbes.com/sites/forbestechcouncil/2018/03/13/why-digital-transformations-fail-closing-the-900-billion-hole-in-enterprise-strategy/#137fa0067b8b

APPENDICES

Appendix 1. State of Agile Survey

First some background

As it is useful to understand how different groups at Fastems see things differently, we have a couple of questions that we seek to cluster the respondents a bit.
These include the part of our organisation you most identify with and how long have you been with us.

What do you do (function and group)?

_____

_____

_____

_____

How long have you been at Fastems

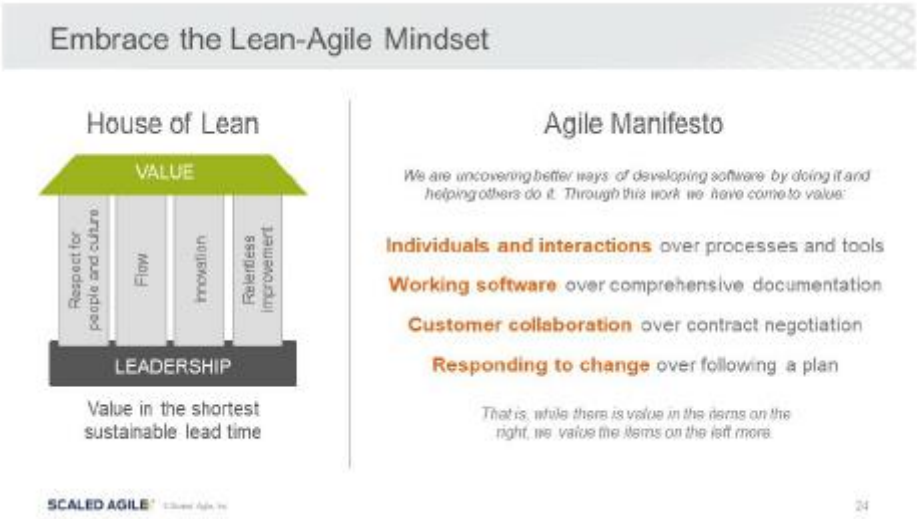
- ◯ Less than a year
- ◯ 1-3 years
- ◯ 4 – 7 years
- ◯ 7 – 10 years
- ◯ 10 - 20 years
- ◯ over 20 years

The following 12 questions will look at the qualities and capabilities of the Lean-Agile companies that we would like to develop at Fastems as well.

We shall try to explain the capability with the answering options and question introductions, also by providing links to relevant articles that you are encouraged to visit should you wish to better understand what the question is all about.

At each point we ask you to estimate where we are in our journey of learning that skill.

Question 1/12: What is the leadership mindset like



The quoted Agile Manifesto and the principles described by the House of Lean are the building blocks of Lean-Agile mindset. Please tell what you think that our leadership's (those that have the power to make decisions) mindset is like.

- ◯ 1. Sit: Leadership mindset is not yet based on Lean-Agile mindset
- ◯ 2. Crawl: Leaders are aware of Lean-Agile principles
- ◯ 3. Walk: Leaders are starting to demonstrate Lean-Agile Mindset and behaviors
- ◯ 4. Run: Leaders advocate, coach and train others in Lean-Agile
- ◯ 5. Fly: Lean-Agile mindset and principles exemplify the culture and influence every decision

If you would like to study these further, there is a lovely article at: https://www.scaledagileframework.com/lean-agile-mindset/

And of course the agile manifesto itself can be found online too: http://agilemanifesto.org/. If you go that far, please don't forget to check the principles behind it.

Question 2/12: Principles

The Lean-Agile mindset is further elaborated for example by the SAFe Principles. Principles are something that should be guiding us especially in our larger scale, but also most other kinds of work. The set of principles we're currently especially interested is:

#1 Take an economic view

#2 Apply systems thinking

#3 Assume variability; preserve options

#4 Build incrementally with fast, integrated learning cycles

#5 Base milestones on objective evaluation of working systems

#6 Visualize and limit WIP, reduce batch sizes, and manage queue lengths

#7 Apply cadence, synchronize with cross-domain planning

#8 Unlock the intrinsic motivation of knowledge workers

#9 Decentralize decision-making

#10 Organize around value

Please consider how these principles have taken flight in our life.

◯ 1. Sit: Principles have not been introduced

◯ 2. Crawl: Leaders are trained in the principles

◯ 3. Walk: Leaders are starting to apply and teach the principles

◯ 4. Run: Challenges and impediments are addressed with the principles

◯ 5. Fly: Principles are foundational and inform most decisions and behaviors

If you feel you want to study them further after replying to this questionaire, you can do that at https://www.scaledagileframework.com/safe-lean-agile-principles/,

Question 3/12: Leading the Change

This questionnaire seeks to showcase where we are in our transformation, in other words changing the organisation. This change can be described in many ways, including John Kotter's famous (and due to some misunderstandings infamous) 8 steps of leading a transformation which are also mapped into some agile transformation strategies. The steps are:

- Establish a sense of urgency
- Create a powerful guiding coalition
- Develop the vision and strategy
- Communicate the vision
- Empower employees for broad-based action
- Generate short-term wins
- Consolidate gains and produce more wins
- Anchor new approaches in the culture

In relation to the above list of steps, how would you say we are doing in leading the change?

- ◯ 1. Sit: Status quo
- ◯ 2. Crawl: Desire to change has reached the tipping point, a sense of urgency for the change exists
- ◯ 3. Walk: Others are leading the transformation
- ◯ 4. Run: Leaders are actively leading the transformation
- ◯ 5. Fly: Leaders are solidifying the new way of working in the culture

If you would like to study the 8 step model and transformation path further, the basic model is introduced at https://www.kotterinc.com/8-st... and their application to agile transformation is discussed in a set of articles available at https://www.scaledagileframework.com/implementation-roadmap/

Question 4/12: Agile Teams

Agile teams are by definition cross-functional, self-organizing entities that can define, build, test, and where applicable deploy, increments of value. This value can be formed of software, software and hardware together, or by something else such as knowledge. Please assess our situation with Agile teams, based on what you know of our organisation.

- ◯ 1. Sit: Traditional, siloed functional development
- ◯ 2. Crawl: Some Agile teams are established
- ◯ 3. Walk: Cross-functional Agile teams apply common roles and practices
- ◯ 4. Run: Agile teams are performing well and flow is established
- ◯ 5. Fly: Agile teams have achieved operational excellence and are continuously improving their approach

One of the best descriptions of agile teams can be found at the Scrum guide: https://scrumguides.org/scrum-... and the SAFe's article on the subject is rather excellent as well https://www.scaledagileframewo...

Question 5/12: Agile Development

Agile development is by definition iterative. It accomplishes its targets by working on the overacrhing deliverables in a set of iterations, improving the result at each pass.

Please assess how well the development work at Fastems sticks to these ideas.

- ◯ 1. Sit: Work is executed in waterfall
- ◯ 2. Crawl: Some teams work in iterations; stories often take multiple iterations to complete
- ◯ 3. Walk: Teams define, build, and test stories in an iteration
- ◯ 4. Run: Teams define, build, test, and deploy stories in an iteration
- ◯ 5. Fly: Teams define, build, test, and deploy functionality multiple times in an iteration

Question 6/12: Built-in Quality

In the agile approach, the built-in quality is a special term. It means putting quality in from the very beginning instead of doing something, testing it and then finding out if it really was proper and working. This term can also be described by one of the Agile Manifesto's principles: Continuous attention to technical excellence and good design enhances agility. One of the key ways to implement it is to follow test led approaches such as TDD, BDD and ATDD, moving testing heavily to the left on the time frame of development efforts

Please consider how Fastems is doing in this aspect.

- ◯ 1. Sit: Quality is poor and inconsistent
- ◯ 2. Crawl: Some testing is done as part of each iteration
- ◯ 3. Walk: Most testing is automated and done within the iteration
- ◯ 4. Run: Teams are applying TDD, BDD, ATDD, architecture, design and release quality practices
- ◯ 5. Fly: Releases are nearly defect free

The concept at its core isn't that complicated, but there is a very large and quite helpful in depth article about its implementation available at:
https://www.scaledagileframewo...

Question 7/12: Organising around the value

How do you feel about the way we're organised, not only on paper, but in our actual work? Have we organised ourselves around value production so that those who take part in producing our deliveries are actually working together?

- ◯ 1. Sit: Functional silos; work is planned centrally
- ◯ 2. Crawl: Agile teams are formed into Teams of Agile Teams
- ◯ 3. Walk: Teams of Agile Teams plan, integrate, release, and learn together
- ◯ 4. Run: Teams of Agile Teams are organized around value and include business and operations
- ◯ 5. Fly: Teams of Agile Teams deliver end value quickly and predictably

If the concept seems alien even after looking at the answering options, the principle on organisation is quite nicely explained here: https://www.scaledagileframewo...

Question 8/12: DevOps Culture

DevOps has been one of the hottest hype words in the software development world for a while. It has also been one of the most misunderstood and misused words. We say this because it has at times been assumed that DevOps refers only to some tools and purely technical practices when in fact, DevOps is closely related to the agile practices in general and a key enabler in what we're trying to do.

How would you rate Fastems' DevOps culture?

- ◯ 1. Sit: Development and operations are functional silos
- ◯ 2. Crawl: Teams are trained in DevOps
- ◯ 3. Walk: Dev and Ops are collaborating on automation and deployments
- ◯ 4. Run: Dev and Ops are integrated and focused on measurement and recovery
- ◯ 5. Fly: Teams of Agile Teams define, build, deploy, release and operate the system

If DevOps is strange to you good place to start besides the Wikipedia (https://en.wikipedia.org/wiki/DevOps) is this article https://www.scaledagileframework.com/devops/

Question 9/12: Continuous Delivery Pipeline

A seminal concept in the kind of DevOps that we are going for is the **Continuous Delivery Pipeline**. It has four parts: The Continuous Exploration, The Continuous Integration, Continuous Deployment and Release on Demand.

The Continuous exploration deals with finding out and clarifying the customer needs we're working with. The continuous integration covers the development activities (including verification and validation of the development results), Continuous Deployment covers the activities that an organisation takes in order to take the development results into staging environment. In this approach the deployment is severed from release which is the last part.

As all of this deals with the release process, we'd ask you to please assess how the Fastems' releases look like when compared to the idea.

- ◯ 1. Sit: Releases are infrequent and problematic
- ◯ 2. Crawl: Some continuous integration; the later steps (like conf) are informed by the previous steps.
- ◯ 3. Walk: Continuous integration and continuous deployment are largely automated
- ◯ 4. Run: Release is decoupled from deployment and available on demand
- ◯ 5. Fly: Continuous exploration drives rapid deployment of innovations

For fuller description of this kind of a pipeline, you can go to the article: https://www.scaledagileframework.com/continuous-delivery-pipeline/

Question 10/12: Scaled Lean-Agile Development

There is only so much that an Agile team can accomplish. When things get too big to handle with one team, the development needs to be scaled up. That means doing it in a larger synchronised set.

How does scaled agile development look like at Fastems?

- ◯ 1. Sit: Traditional, waterfall, stage-gated system development
- ◯ 2. Crawl: Some Teams of Agile Teams formed
- ◯ 3. Walk: Coordination exists across Teams of Agile Teams and suppliers
- ◯ 4. Run: Horizons of planning, solution roadmaps, and Agile contracts guide work
- ◯ 5. Fly: Large programs meet milestones reliably and with quality

Question 11/12: Combining more demanding blocks together

When we have to deal with either larger than normal software systems or with systems that we refer to as cyber-physical systems (Systems that deal with both hardware and software), then we face new challenges and for example the integration can no longer remain truly continuous, but needs to become continuish.

There are also other things we need to consider while trying to rapidly deliver these systems in fully functional form. Fitting the software and hardware together we are engaging in solution integration and this can be done in an agile fashion as well.

So how do you think we're doing at this level of agility?

- ◯ 1. Sit: Integration of the solution is late and problematic
- ◯ 2. Crawl: Frequent integration of software elements; infrequent solution integration
- ◯ 3. Walk: Continuous software integration and regular full solution integration
- ◯ 4. Run: Full or partial solution integration every iteration
- ◯ 5. Fly: The solution always runs

If your curiosity was piqued by this question more information on application of agile practices at this level can be found here: https://www.scaledagileframework.com/enterprise-solution-delivery/

Question 12/12: How are we exploiting variability

Let's face it. When we're trying to deal with complex systems and complex needs, the variability is pretty much certainty. Therefore requirements need to be handled differently when we're playing the agile game. There are different ways for dealing with this variability and exploiting it is a one of these options. One we're really aiming for.

Please consider which of the below alternatives best describes how we're handling requirements and variability.

- ◯ 1. Sit: Requirements and design are fixed up-front
- ◯ 2. Crawl: Some requirements and design are variable
- ◯ 3. Walk: Model-based systems engineering anchors development
- ◯ 4. Run: Set-based design explores and validates alternatives
- ◯ 5. Fly: On-time solutions deliver best economic outcomes

If you happen to be interested in learning more of how to exploit variability preserving options, there are couple of rather interesting approaches described in an article at https://www.scaledagileframework.com/assume-variability-preserve-options/

One more thing

Thanks for answering all the predefined questions of this questionaire.
Now the floor is yours. If there is anything else you would like to comment or suggest about the agile transformation, now you get your chance for it.

_____

_____

_____

_____