A Case Study on Streamlining Workflows Between Product, Engineering, and QA in a SaaS Environment (Company X)

Submitted By:

Taskeen Haider (2403722)

UNIVERSITY OF VAASA

Masters in Strategic Project Management

UNIVERSITY OF VAASA

Author: Taskeen Haider

Title of the Thesis: A Case Study on Streamlining Workflows Between Product, Engineering, and QA in a SaaS Environment (Company X)

Degree: Master of Science

Programme: Strategic Project Management

Supervisor: Marko Mäkilouko

Year: 2026

Pages 70

Abstract

In modern Software-as-a-Service (SaaS) companies, cross-departmental workflows play a vital role in delivery of quality and reliable products. The SaaS environment is typified by a high rate of customer expectation change and the necessity of having development teams continuously modify their processes to accommodate that change and be fast and quality-oriented. As a result, the ability to align the work among the departments, such as Product Management, Engineering, and Quality Assurance (QA), becomes one of the most important organizational capabilities. Research has found that cross-functional collaboration is a prevailing aspect in the production of dependable and consumer contentment in the software development environment (Dingsoey et al, 2012). This thesis evaluates and analyzes the opportunities, with the aim of continuing to enhance the processes of Product, Engineering and Quality Assurance departments in Company X, a business management software company in the wellness and fitness business. The experiment focuses on how communication, collaboration practices and workflow transparency affect the efficiency of product development processes. Rather than targeting such key failures in operations, the study looks at how an already operational system can be reviewed and even enhanced upon by taking systematic process analysis. The study employs the qualitative case study methodology so as to gain insights into the working process within the organization. The data is gathered by using semi-structured interviews with the stakeholders and internal documentation and direct observation of workflow processes at the company. Moreover, this paper uses Kaizen (continuous improvement) and Value Stream Mapping (VSM) as the frameworks to examine and visualize the current development workflow. These frames can be used to recognize small flaws and areas of enhancement and retain the merits of the current system (Rother and Shook, 2003). VS Mapping enables the visualization of the overall process of the workflow starting with reception of a customer request to the very end of the issuance of a product improvement. The analysis of the value-adding and non-value-adding wait

times offers a better insight into the flow of work across the departments. Concurrently, Kaizen focuses on gradual changes and urges companies to constantly enhance their processes as opposed to pursuing radical structural transformations (Womack and Jones, 2003). The anticipated result of this study is a list of practicable recommendations and visual process maps that can promote the culture of constant enhancement across the departments. In terms of academics, the thesis will help comprehend how Lean improvement techniques, including Kaizen and Value Stream Mapping, can be implemented in SaaS-based organizations where the main working process is information-sharing, not a manufacturing process. The results reveal the ability of the structured workflow assessment to assist software businesses in sustaining effective cooperation as they scale their business and development teams.

Keywords:

SaaS workflow, Value Stream Mapping, Kaizen, Lean thinking, cross-functional collaboration, process improvement, Agile development, workflow efficiency

Table of Contents

Chapter 1: Introduction

1.1 Background

Software-as-a-Service (SaaS) organizations get to play in a very dynamic environment where the product development has to be able to respond fast to the needs of the customers and be both stable and reliable. SaaS platforms, unlike the traditional models of software development, need constant updates, the increase in features, and the elimination of bugs. Such a cyclical process of enhancement demands tight cooperation among various departments, especially Product Management, Engineering, and Quality Assurance (QA). How effective this collaboration is directly affects the speed of release, the quality of the product, and customer satisfaction (Sommerville, 2016).

SaaS companies today are counting on cross-functional teams to organize development efforts. Product managers establish the requirements, and rank features according to business strategy and customer feedback. Software development is used by engineering teams to provide the necessary functionality, and QA teams ensure that new functionality is working and meets the specified acceptance criteria. Other teams like support or technical operations within an organization are also involved in this workflow to identify user issues and make improvement requests in most organizations. Due to the varying responsibility and outlook of these teams, communication and coordination are vital to the development so that the development process can be smooth (Beck et al., 2001).

Company X is a SaaS firm offering an end-to-end business management platform to wellness, fitness and service-based businesses. The platform is a combination of various important functions such as scheduling, billing, customer relationship management (CRM), marketing automation, and reporting features. The system is used by clients to maintain the day-to-day running, customer communications, and business performance. With the expansion of the platform and the introduction of new functionality, it becomes even more essential to ensure a productive product development process.

The organization has a step-by-step development process with the help of various collaboration tools. Jira Tickets and task tracking: Jira is employed to manage the ticket and track tasks, Confluence Documentation and knowledge sharing: Confluence is adopted to document and share knowledge, Figma Product design and user interface prototyping: Figma is utilized to design products and prototyping user interfaces, and TestRail QA verification and testing processes: TestRail is employed to track and verify the quality of the tasks. They are employed to design communication between teams and provide clarity to the development processes. However, even with the apparent tools and procedures, the workflow has to be continually monitored and optimized.

In Company X, customers can make feature requests or report problems to the Support team, which can kick off product improvements. Technical Support Engineering (TSE) department then reviews these requests to gauge whether they are legitimate product enhancements. After validation, the requests are forwarded to the Product team to be prioritized and analyzed. Before product managers define detailed specifications to be followed in development, they may obtain more information about the customers by conducting user interviews or product usage analysis.

Once the requirements are defined, the design team develops visual mock-ups with Figma to show how the user interface and functionality should look like. After completing the design process, the Engineering team reviews the request and makes technical estimates in story points. The ratio of story points to engineering work in a day is about one to one day in the development model of the organization. Once estimation is done, development work is started and engineers add the necessary feature or enhancement.

Once development is complete, the feature is transferred to the Quality Assurance phase where a set of QA engineers will make the implementation undergo testing to ensure that the implementation is in accordance with the specified requirements. In case problems are found in the course of testing, the feature is sent back to the Engineering team to fix. After successful testing, the Product team carries out a final validation and then the improvement is added to the next release cycle.

Although such a formalized process plays a role in the cooperation of different departments, the dynamics of SaaS developmental environments suggest that insignificant inefficiencies may emerge over the long-term. These inefficiencies might be in the form of minor delays in communication, re-explaining processes, or unneeded validation mechanisms among teams. Such issues do not necessarily imply that the system has broken down but are merely a part of the challenges related to organizing work and collaboration with distributed teams and work across time zones.

This is why organizations should frequently assess their workflows and find ways to improve them. Kaizen and Value Stream Mapping are continuous improvement models, and analytic tools that offer a systematic way of examining workflows and finding points of efficiency that are able to be increased without interfering with the entire system.

1.2 Research Motivation

This thesis is motivated by real-life experience of working in the Product Operations department, Company X. Daily, there is coordination between Product, Engineering, QA, and Technical Support Engineering in this role. These interactions are valuable in understanding the process of the development workflow in action and identifying areas in which the process can be improved.

By daily participating in product operations, it was clear that, although the system is somewhat functioning well, certain workflow steps can be more transparent and communicated. As an example, during the development process, there are cases when the Engineering teams would like to have more explanation of the feature requirements, or when the results of the QA testing would need more explanation by the Product managers. This kind of interaction is typical of complex development processes, but sometimes gives rise to minor delays when there is unavailability of information.

Contemporary communication tools like Jira, Slack, and Microsoft Teams can be used to aid in communication among teams, although the efficiency of these tools largely relies on their usage within the organization. With the help of advanced tools, the communication gaps can still take place in cases where the processes are not regularly adhered to, or the expectations between teams are not agreed upon.

This research is thus not aimed at defining any major issues within the organization but is an attempt to observe how a system that is in operation can be enhanced and maintained as the company expands. Continuous improvement models like Kaizen make organizations to review their processes after every short period, as well as bringing minor changes which enhance efficiency in the long run. Implementation of these principles into the Company X workflow gives a chance to reinforce collaboration practices and ensure operational efficiency as the organization grows.

1.3 Theoretical and Practical Relevance

Theoretically, this research can be relevant to the research on Lean management and Agile development by studying how continuous improvement models could be implemented in the digital product setting. The idea of lean like waste minimization and workflow optimization was initially

worked out in the areas of the manufacturing industry where value creation is the process of physical production. Yet, in software development settings, value is generated by exchanging information, communicating, and engaging in knowledge work as opposed to material resources.
This distinction implies that Lean tools should be adjusted in the case of applying them to SaaS companies. Software development processes typically exhibit waste as communication delays, lack of clarity in requirements, unjustified approvals or redundant test cycles instead of material inefficiencies. This paper will help to comprehend how Lean thinking can be applied to the present-day software development setting by implementing Value Stream Mapping to digital workflows (Poppendieck & Poppendiek, 2003).

Practically, the study has a wealth of information to organizations that aim to sustain effective development procedures in the process of expanding their operations. The recommendations created during this thesis are designed to enhance the cooperation between Product, Engineering, and QA teams and retain the current strengths of the Company X development process. Instead of suggesting some major structural modifications, the research aims at small-scale enhancements that can facilitate the overall efficiency of operations.

1.4 Research Objective

The main aim of this paper lies in the assessment of the maturity of the present workflow between Product, Engineering, and Quality Assurance departments of the Company X and the opportunity of streamlining the processes through the concepts of Kaizen and Value Stream Mapping. The study will enhance communication, increase transparency in the workflow, and long-term operational efficiency across departments.

1.5 Structure of the Thesis

This thesis is categorized into five main chapters.
Chapter 1: This chapter introduces the research background, motivation, and objectives of the study.
Chapter 2: This Presents the literature review and theoretical framework, focusing on Agile development, Lean thinking, Kaizen, and Value Stream Mapping.
Chapter 3: This chapter explains the research methodology and describes the qualitative case study approach used to collect and analyze data.
Chapter 4: This chapter presents the empirical findings and evaluates the current workflow using Value Stream Mapping.

Chapter 5: and finally, this chapter provides analysis and discussion of the results, followed by final conclusions and recommendations.

1.6 Research Questions

Main Research Question:

"How can continuous improvement frameworks such as Kaizen and Value Stream Mapping be used to evaluate and enhance cross-functional workflows between Product, Engineering, and QA at Company X?"

Sub-Questions:

1. What are the current strengths and best practices in the workflows between Product, Engineering, and QA?
2. Where do minor inefficiencies, communication gaps, or redundant steps occur in the existing processes?
3. How do tools such as Jira, Confluence, and Slack support or limit transparency and alignment between departments?
4. How can Kaizen and Value Stream Mapping help visualize, measure, and continuously optimize workflows?
5. What recommendations can help maintain collaboration efficiency as Company X continues to scale?

1.7 Research Gap

Despite the fact that Agile development and Lean management are thoroughly examined in academic literature, only a few studies are aimed at the evaluation of already functioning workflows. The bulk of process improvement studies is focused on addressing big operational blowups or inefficiencies instead of keeping the best in developed systems.

Moreover, there is a lack of studies that combine Product, Engineering, and QA processes in the SaaS companies. The addition of the roles like Technical Support Engineering adds an extra layer of complexity to the workflow and gives a chance to learn how different organizations align the improvement requests with the work of multiple teams.

Besides, the integrated application of Kaizen and Value Stream Mapping in SaaS developing settings is under-researched. Although these techniques are well developed in manufacturing sectors, their usage in the digital products development processes is still in its infancy in the scholarly literature. In this way, this study adds to both theoretical and practical knowledge of the applicability of Lean improvement techniques to software development organizations.

1.8 Expected Contribution

The research study contributes practically and theoretically to the literature on workflow optimization in SaaS companies. Regarding practicality, the thesis provides a comprehensive analysis of the interdepartmental operations within Company X, and identifies a means of enhancing the operations within the Company in small steps. Theoretically, the article extends the application of Kaizen and Value Stream Mapping to the digital product development contexts where a workflow optimization depends on communication and information transfer rather than on the actual production processes and fast SaaS providers where it is crucial that efficient and open workflow across different departments is fundamental to the development of high-quality products.

The thesis evaluates and analyzes the spheres of additional improvement in the procedures of Product, Engineering, and Quality Assurance (QA) teams of Company X, the business management software company, which operates in the wellness and fitness industry. This paper will examine the contribution made by collaboration, communication and openness of the process to the successful delivery of products and how they can be sustained and even improved over time.

The qualitative case study approach will be used to collect the data, which will be in the form of interviews, internal documentation review and process observation. Kaizen (continuous improvement) and Value Stream Mapping (VSM) are the guiding principles used in the paper to map the current process, identify marginal inefficiencies, and where value creation can be reinforced. This paper is not aimed to cover the major problems of the organization, rather it is aimed to review the current degree of maturity of the workflows and suggest how to further enhance the degree of alignment, decrease in communication and cross-team performance as the company is growing.

The projected result is a series of realistic recommendations and process maps, which will support a continuous improvement attitude among the departments. On the academic front, the study can help to understand how Kaizen and VSM can be successfully implemented in SaaS companies to analyze workflow performance, enhance team collaboration, and facilitate strategic project management practice.

Chapter 2: Literature Review and Theoretical Framework

2.1 Introduction to the Literature Review

This chapter provides the theoretical base of the thesis through a review of the current academic literature on the related topics of cross-functional collaboration, workflow management, and continuous improvement in the software development settings. The aim of this literature review is to come up with a conceptual framework that facilitates the empirical exploration of workflows between Product, Engineering, Quality Assurance (QA) and Technical Support Engineering (TSE) at Company X, a Software-as-a-Service enterprise that is in the wellness and fitness software market.

Since this research aims at assessing and optimizing workflows in a SaaS company, the literature review is based on several streams of researches such as project management, Agile and Lean methodologies, process improvement theories and cross-functional communication. These areas of research provide the theoretical background that can be applied to research the workflow performance and collaboration in an environment that is knowledge-intensive. A special focus is put on Kaizen and Value Stream Mapping (VSM) which are the main analytical models that will be applied subsequently in the thesis to analyze the existing workflow and find the areas where it can be improved. The concepts were applied in manufacturing industries but have been more adjusted to the environment of service and digital manufacturing where work relies on information exchange and is not based on physical production (Dingsoyr et al., 2012).

The chapter is clearly divided into a number of parts in order to establish an effective theoretical framework. To begin with, the importance of project management within software and SaaS settings is addressed to give a background surrounding the coordination of workflows in the development of digital products. Second, Agile practices are considered as the popular development model in contemporary SaaS organizations. Third, Lean thinking is presented as a more general management philosophy aimed at efficiency and minimizing of waste. Fourth, Kaizen is discussed as a philosophy of continuous improvement that focuses on process refinement in a small stepwise manner. Fifth, Value Stream Mapping is touched upon as an effective tool to visualize and analyze workflows. Lastly, the chapter examines the literature on cross-functional collaboration, communication practices, workflow transparency, and role clarity in software development teams. Combined, these opinions constitute the theoretical framework of the analysis and improvement of the workflows in Company X.

2.2 Project Management in Software and SaaS Environments

Project management plays a very important role in the planning of work, allocation of resources and presentation of results in a timely manner in very complex organizations. The classical project management within the industries can be structured in such a way that the projects are completed in a specific process which has specific phases such as planning, execution and closure. However, software development projects and SaaS projects in particular are very unpredictable, volatile and constantly change products. Such characteristics make the traditional linear models unsuitable to manage modern software development processes (Rother and Shook, 2003).

SaaS organizations operate within an environment in which the product development is both a process and not an endpoint. SaaS products are continuously being developed and improved through an iterative cycle of development as opposed to the traditional projects which have specific start and end points. Frequent updates of new features, improvements and bug fixes are made based on customer feedback, market trends and technological changes. The dynamic nature of the environment requires project management practices which would enable flexibility, rapid decision making, and close collaboration with other teams. Scholars have indicated that the current software project management is less about planning and documentation and more about coordination, communication and flexibility (Womack and Jones, 2003).

The project management in a SaaS environment oversteps such traditional project management concerns as scheduling and budgeting. It involves also the coordination of team dependencies, synchronisation of priorities across departments and the flow of information between stakeholders in an efficient manner. The Product Managers define the requirements and priorities on how it should be developed based on customer value and business strategy. Engineering teams translate these requirements into technical solutions and they are verified to assure that the features developed are of quality and functionality as expected by the QA teams. Support services such as Technical Support Engineering may also contribute to this by validating customer issues and providing a feedback of real application of the system.

With such a complex interdependence of functions, project management in SaaS organizations must pay attention to both the technical and organizational aspects of development. Such process as the coordination, the communication of teams and goal alignment influences the effectiveness of the working process in the development in general. That is why, many researchers call software project management a socio-technical system, where technology, people and organizational processes are closely intertwined (Sommerville, 2016). This belief underlines that the importance of studying workflows is not as technical processes but as interactive processes and they are affected by patterns of human interaction and communication.

The present study employs this socio-technical method in examining the workflows in Company X rather than examining the amount of tasks performed, and the speed of the development process, the study examines the impact of communication practices, the coordination and transparency of the process on the effectiveness of the workflow. The methodology warrants the use of qualitative research methods and process-based models such as Kaizen and Value Stream Mapping that emphasize studying systems as a whole, rather than as individual activities.

2.3 Agile Methodologies in Software Development

The Agile, as methodologies, have been the preferred model to control software development projects, especially in SaaS organizations where agility and speed are paramount (Highsmith, 2002). The agile approaches were created as a reaction to the shortcomings of the conventional project management methodology which was based on extensive initial planning and strict stages of development. These conventional procedures frequently failed to respond to the requirements and client feedback in the dynamism of software-based settings. Published in 2001, the Agile Manifesto came up with a new philosophy which focused on people and interaction, working software, customer interaction, and responsive to change (Beck et al., 2001).

Agile models like Scrum and Kanban organize the development process into short development iterations, commonly known as sprints (Sutherland, 2014). In such cycles, teams map out work, deliver features, test functionality and evaluate outcomes and then start the next cycle. The ceremonies encompassed in Agile are daily stand-ups, sprint planning, sprint review, and retrospectives. The activities foster transparency, facilitate communication and offer teams with chances to perpetually review and refine their work processes (Poppendieck & Poppendieck, 2003).

Theoretically, Agile frameworks can ensure close cooperation between Product, Engineering, and QA teams as they promote a sense of collective responsibility towards product outcomes. Product managers create user stories and acceptance criteria which formulate development. These features are put in place by engineers without compromising the technical feasibility and system stability. Before the software is released, the quality and functionality of the software is checked by the QA teams. These roles are expected to communicate frequently in order to minimize misunderstandings and allow fast adaptations in the development.

Nevertheless, there are studies that indicate that the implementation of Agile practices fails to automatically remove workflow issues. Although Agile frameworks offer an outline on how teams can work together, their efficiency relies on the way teams perceive and apply them to their unique organizational setting. The disparities in priorities, communication, and technical understanding may continue to cause differences between the departments. Product teams can work to maximize customer value, Engineering teams can work to guarantee technical feasibility and system performance, and QA teams can work to guarantee risk mitigation and defect prevention. Such variations may cause communication failures or delays when expectations are not clearly established (Hines et al., 2004).

Agile practices in SaaS organizations are usually made to suit the requirements of the organization and not necessarily applied to a set of frameworks. Studies indicate that hybrid organizations that mix Scrum, Kanban, and Lean practices are widely practiced in reality to promote flexibility and continual delivery (Schwaber and Sutherland, 2020; Rother and Shook, 2003). This flexibility is an indication of the dynamic nature of software development environments where the changing customer needs and constant product evolution necessitate flexible process structures. Consequently, Agile practices can be interpreted as company habits that are integrated into the work processes and are not strict structures that have to be adhered to.

2.4 Lean Thinking in Knowledge Work

Lean thinking was first developed in the manufacturing sectors, and was mostly applied in the Toyota Production System, where it was applied to enhance production efficiency and minimize unnecessary wastes. The fundamental concept of Lean is to maximise customer value and tone down those activities that are not related to value. The main ideas of lean are to find the value in the eyes of the customer, value stream mapping, development of smooth flow of processes, production with the help of the pull and constant movement towards perfection by means of the continuous improvement (Schwaber and Sutherland, 2020).

Although Lean thinking was originally developed for physical manufacturing environments, many researchers have argued that its principles are highly applicable to knowledge-intensive work such as software development. In knowledge work environments, value is not created through the transformation of raw materials but through the creation, processing, and communication of information. As a result, waste takes different forms compared to manufacturing. Waste can be in the form of communication delays, physical products that are defective or excess inventory instead of having to move surplus inventory or defective physical products (Ohno, 1988).

Such inefficiencies are especially susceptible to software development workflows as much of the work relies on the cooperation of more than one role and department. In case of information incompleteness or misinterpretation at any point of the working process, time wastage and the need to clarify the matter several times may arise. Lean thinking is an approach that urges organizations to think of workflows as a whole instead of trying to optimize individual tasks separately. Such system level view assists the organizations to realize any form of inefficiencies that arise at the interface points of the teams.

Uses of Lean in SaaS environments tend to be through reduction of waiting between development cycles, reduction of work in progress, improvement of feedback loops and development of visibility in task management. These practices come in handy in making sure that the work flows through the departments without any unwarranted interruptions and delays. As an example, a development queue can prevent simultaneous work on a large number of tasks to reduce multitasking and enable teams to better complete high-priority features.

The applicability of Lean thinking to this research is the focus on the analysis of workflows as interdependent systems. The workflow between Product, the Engineering, QA, and Technical Support Engineering in the context of the Company X is a value stream under which the customer value is provided in the shape of product improvements and bug fixes. By comprehending the functioning of this value stream, it is feasible to recognize the chances of minimizing the inefficiencies and enhance teamwork.

2.5 Kaizen: Continuous Improvement Philosophy

Kaizen is a Japanese management philosophy that is translated as change to better or continuous improvement. The concept underlines the notion that organizational processes can be continually enhanced without making massive changes but implementing small, incremental ones. In contrast to traditional change management techniques that presuppose restructuring or significant changes periodically or after major reforms, Kaizen promotes reflection and slow advancement during daily work (Anderson, 2010).

The participation of the employees throughout the organization is a major tenet of Kaizen. Rather than having top-down decisions on management, Kaizen will encourage people working in the process to spot inefficiencies and suggest ways to improve. As these people are in a daily contact with workflows, they are usually in a good position to appreciate small inefficiencies, which may not be noticed by the management. Kaizen allows organizations to promote a culture where improvement is a routine undertaking through involvement and experimentation.

Agile practices such as retrospectives, sprint review, and continuous feedback loops, which may be found in software development, can be regarded as the application of Kaizen principles. Teams in retrospectives retrospectively examine what has worked well in a sprint and what they would prefer to do when creating the process in a better way. The outcomes of such discussions are often minor modifications in the processes, modes of communication or norms of development. All these small contributions with time will result in significant changes in the overall efficiency of workflow.

However, even when lots of improvement is discussed in software teams, it is usually within the scope of individual teams. An example is that development teams may operate on practices in code or internal task coordination without taking into account bigger cross-functional processes in the Product, QA, or support teams. This means that certain inefficiencies interdepartmentally may go unchecked. Implementing Kaizen concepts in the workflow level helps organizations to analyze the interactions between various teams and where communication or coordination could be enhanced.

Kaizen can be considered as the philosophical basis of the analysis of the workflows in Company X instead of assuming the current process to be imperfect, the analysis of the workflow is viewed as the possibility to make an already operating system even stronger. The organization will be able to promote collaboration by finding little areas of improvement, e.g. by making communication practices more transparent or having a clearer definition of the handoff process, without disrupting the development process.

2.6 Value Stream Mapping in Software and SaaS Contexts

Value Stream Mapping (VSM) is a Lean management concept that is utilized to map the process of activities needed to provide the customer with value. VSM was initially applied in manufacturing settings where it was applied to visualize material flows and information flows in production systems. The tool is meant to determine the value-adding and non-value-adding activities, the waiting time, and inefficiencies in a workflow. By rendering these elements visible, organizations are in a better position to appreciate the flow of work through a system and where it can be enhanced (Reinertsen, 2009).

Over the past few years VSM has been applied more to service industries and knowledge-based workplaces such as software development. In software terms, a value stream is the sequence of actions that occurs between the time a customer request or need has been defined through to the delivery of the appropriate feature or fix to the users. This process generally consists of a requirement analysis phase, design, development, testing, validation and release.

Among the key advantages of VSM, it is the possibility to determine the inefficiencies that could not be observed when considering the single tasks. To illustrate, every department in an organization could work effectively in its area of responsibility. Nevertheless, there can still be delays when a work is transferred between various departments. These delays are common at handoff points where there have to be transfer of information among teams. In case the information is not complete or understandable, some more communication just might be needed before the following step of the process can be started.

In cross-functional environments such as SaaS development organizations, these handoff points are common. Product managers must communicate requirements clearly to engineering teams. Engineering teams must ensure that their implementation meets the expectations defined by Product. QA teams must verify that the functionality works correctly and meets quality standards. All these transitions are some of the points where misunderstanding or delays can be experienced.

Value Stream Mapping is another tool to work out this problem, as it allows developing a graphic illustration of the whole work process. Mapping every step of the process and measuring the waiting times and process times related to it, organizations are able to see where value is generated and where there are delays that can be avoided. VSM as an analytical tool and as a communication tool is employed in this thesis. It allows visualizing the existing workflow in Company X and helps to develop a future-state workflow that minimizes the inefficiencies and conserves the benefits of the current system.

2.7 Cross-Functional Collaboration and Communication

Cross-functional collaboration is defined as the harmonious interaction of teams of varying expertise who collaborate towards the same goal. In software development settings, cooperation among Product managers, software engineers, QA specialists, and support-related positions are necessary in achieving workable and dependable products. All these roles bring in a new dimension to the development process and proper cooperation will see that these dimensions are put together in an effective manner (Liker, 2004).

Studies on software development organizations have always reported communication issues as one of the most important factors that influence workflow performance. The impact on the interpretation of information and the expression of expectations can be caused by differences in professional background, technical expertise, and priorities among teams. Product teams can be customer-centered and business-value-oriented, engineering-centered and system performance-oriented, and QA-centered and defect-detection-based risk mitigation. In the absence of good communication channels, such differences in priorities can result in miscommunication or delays.

To overcome these issues, organizations tend to use common artifacts and collaborative tools where these facilitate communication and documentation. Structured communication mechanisms that facilitate team members to understand work being done are user stories, acceptance criteria, technical documentation, and testing reports. Issue tracking systems, collaborative documentation systems, and communication software are some of the tools that enable teams to monitor the progress and flow of information more efficiently.

There is however no guarantee of effective communication by the mere existence of collaboration tools. The success of these tools heavily relies on their regular use and the clarity of responsibilities in the workflow. Incomplete documentation or lack of clear communication of expectations leads to teams wasting more time in clarifying requirements or going over what has been done.

This thesis makes cross-functional cooperation one of the key elements of workflow efficiency. Examining the process of communication between departments in Company X and determining the possibility of its improvement, the study will help to better understand how collaborative practices impact the workflow performance in SaaS companies.

2.8 Workflow Transparency and Handoffs in Software Development

Workflow transparency means the extent to which process activities, responsibilities and progress can be seen by the entire participants in a workflow. Transparency is very important in complex development settings to facilitate the coordination and to lessen uncertainty among teams. Because the status of tasks will be readily visible to the stakeholders and they will be able to monitor the flow of work through the system, it will be easier to recognize potential delays and proactively deal with them (Reinertsen, 2009).

In SaaS organizations, development processes usually have multiple handoff locations whereby work moves between work teams. As an example, some of the requirements can shift to Product managers, some completed code can shift to QA to test, and some verified features may shift to Product to be finalized before release. All these transitions entail information transfer between teams, and effective communication is vital to ensure the efficiency of the workflow.

Studies have revealed that handoffs are one of the main contributors of inefficiencies in knowledge-based work processes. Incomplete or ambiguous information in the process of a handoff might mean that the information should be rephrased or further explanation sought by the receiving team. This added communication also brings delays and can cause frustration among teams. There are also instances where work can be reworked by going back to the previous processes of the workflow.

These challenges can be addressed through transparency tools, like shared workflow boards, documentation tools, and process diagrams, which allow all stakeholders to see the workflow. Mapping the workflow enables the teams to visualize where tasks are situated in the process and what needs to be done to get the tasks moving.

Value Stream Mapping helps to improve the transparency of the workflow since it depicts the whole development process in one visual artifact. VSM can help organizations determine the point at which handoffs take place and where communication might need to be enhanced by showing how the work flows between departments. Workflow transparency is considered in this thesis as a system level property that arises out of process design and communication practice in company X.

2.9 Role Clarity and Responsibility in Cross-Functional Teams

Role clarity can be described as the degree to which individuals and teams are aware of their roles in a workflow. In cross-functional setups, it is easy to avoid confusion in terms of ownership of tasks and decision making power because of role definition. By having well-defined responsibilities, teams are able to coordinate their activities and minimize redundant duplication of work (Poppendieck & Poppendieck, 2003).

The roles of Product Manager, Software engineer, QA engineer and Support engineer frequently have close interactions in the software development processes within organizations. Although a certain level of flexibility is a good quality in a team setting, too much ambiguity may result in inefficiencies. As an example, when it is not clear who is supposed to validate a requirement or generate a bug report teams can end up focused on clarifying roles instead of making any progress on the task.

Agile systems also seek to tackle this issue by establishing roles and responsibilities in development teams. Nevertheless, with the growth of organizations and more complex workflows, new roles can arise that can influence the development process. These roles may also bring new coordination issues when their roles are not well incorporated into the working process.

Kaizen helps organizations to work on role clarity by reflecting and making changes in a continuous manner. As the workflow changes, teams do not have to assign roles strictly, but can adapt them over time. This adaptable style promotes collaboration and at the same time, there is clear ownership of every stage of workflow.

The role clarity is considered in the framework of this research as a significant variable that can affect the working efficiency of Company X since the distribution of responsibilities among Product, Engineering, QA, and Technical Support Engineering is evaluated, and the research identifies potential areas of improvement to enhance the effective use of coordination and the reduction of redundant communication channels.

2.10 Information Flow Versus Material Flow

One of the key distinctions between software development and manufacturing is the type of value being created. Value is created in manufacturing settings as physical materials are transformed into the finished products. Conversely, the production of value in software development is mainly achieved through processing and sharing of information. This difference implies much in the application of Lean tools like Value Stream Mapping to the software environment (Humble and Farley, 2011).

In software processes, inefficiency can be generated by piecemeal, tardy and misinterpreted information. Loss of requirements, ambiguous acceptance criteria and lack of information about bugs are all types of information waste that may derail the development process. These problems do not necessarily emerge on the surface but can greatly slow down development and the overall workflow efficiency.

Scientists have thus indicated that the concepts of Lean have to undergo some changes to emphasize the flow of information in the case of software development processes. A software development workflow may be considered to be a sequence of transformations of information. Product requirements are based on customer needs, code is based on requirements, and tested functionality is based on code.

Workflow stage handoffs include transfer of information between teams. In case this information is missing or inaccurate, there can be delays as teams consult or further verify. The flow of information through the work process can be used to identify areas where communication can be enhanced by organizational management to minimize delays and enhance coordination.

Applying Value Stream Mapping to the processes of Company X, this thesis takes an information-flow approach to the analysis, rather than considering the physical processes. The analysis is conducted on the information flow between Product, Engineering, QA, and Technical Support Engineering teams. This point of view justifies the qualitative case study method and assists in determining the patterns of communication that can impact the efficiency of the working process.

2.11 Project Governance and Cross-Department Coordination

Project governance can be defined as systems, structures and decision making mechanisms employed to direct projects and accountability in an organization. Traditional project management settings tend to be characterized by formal reporting lines, fixed approval levels, and decision making is centralized. Nevertheless, the governance in Agile and SaaS organizations is more likely to be decentralized and less rigid due to the necessity to adjust the development processes to the evolving needs and customer feedback (Boehm and Turner, 2004) (Kim et al., 2016).

Governance is also significant in the software development processes where coordination of work between various departments is involved. The priorities are established by product managers according to the business strategy and customer needs, technical feasibility of the proposed solutions is determined by the engineering teams, and QA teams make sure that the implemented functionality is of high quality. Governance mechanisms assist in aligning such views and in making decisions that are consistent throughout the organization.

Co-ordination between departments is most essential in SaaS organizations due to the fact that product development is not done in one team. Rather, it is an ongoing collaboration among various functions that provide various kinds of expertise. The absence of proper coordination tools can result in the isolated activities of departments with unfocused priorities or slowed decision-making processes.

In Agile organizations, modern approaches to governance are based on transparency and mutual understanding as opposed to strict control mechanisms. The visual management tools, which include task boards, workflow diagrams, and process maps, enable the stakeholders to monitor the development activities progress and determine the dependencies among team members. These tools facilitate decision making processes as they enable visibility of the workflow to all users.

In Company X, governance is integrated into the work flow by having formal decision points like requirement approval, development estimation, testing validation and release planning. This study can help us understand how governance practices affect workflow efficiency in SaaS settings by examining how these decision points are handled and communicated among departments.

2.12 Continuous Improvement Culture in SaaS Organizations

A continuous improvement culture is a culture in an organization whereby employees are eager to get a chance to eliminate non-value adding components to processes, enhance teamwork and boost performance (Mann, 2005). Such a culture is especially important to retain in SaaS organizations as products are constantly changing, and the development teams should adjust to new technologies, customer feedback, and market conditions. Companies that promote the idea of constant improvement tend to be in a better position to react to these changes and be competitive in the long term (Forsgren et al., 2018).

Continuous improvement cultures integrate learning and reflection as a part of daily work. Organizations incorporate improvement discussions in the routine development practices instead of viewing improvement as a distinct activity that is conducted periodically. The Agile practices encourage this attitude by introducing such mechanisms as retrospectives, sprint reviews, and feedback loops that enable teams to review their performance and modify workflows.

However, continuous improvement activities are not always aimed at the entire workflow of the organization and are rather oriented towards particular teams. As an illustration, development teams can focus on supporting their code practices or testing policies but fail to identify inefficiencies that take place during interdepartmental handoffs. The solution to the inefficiencies of the workflow is a more generalized view that looks at the interactions of various teams within the system as a whole.

Kaizen offers theoretical basis to the development of a culture of continuous improvement since it focuses on gradual change, as opposed to radical change. Kaizen assists organizations by encouraging their employees to find minor ways to improve their operations in their day to day work without causing the processes to be shaken up radically.

The culture of continuous improvement is especially applicable to Company X since the organization is a fast-evolving SaaS business, where business processes need to change with the development of the product. By analyzing the current development process using Kaizen and Value Stream Mapping, the company can pinpoint areas in which the company can enhance its processes and procedures without altering the stability of the already existing processes.

2.13 Integration of Theoretical Concepts

The theoretical insights discussed in this chapter are combined to make up the analytical framework that will be applied in this study. Both theoretical ideas can be used to provide a particular viewpoint of the workflow efficiency in SaaS organizations. Agile techniques are the means that offer the working environment where the development activity is structured and implemented. Lean thinking brings in a system-level approach that is aimed at efficiency of flow and minimizing waste. Kaizen is a philosophy of continuous improvement that focuses on gradual improvement of processes. Value Stream Mapping is an effective method of visualizing workflow and identifying inefficiencies.

The dimensions of workflow effectiveness that are significant in this framework are cross-functional collaboration, role clarity, transparency of workflow, and flow of information. The thesis does not focus on these factors separately but assumes a complex approach to these factors as they interact. As an illustration, a better role definition can lessen the time taken in communication, and greater transparency can enhance teamwork.

Such a combined structure enables the research to examine the workflows in Company X through the prism of several perspectives at a time. The framework is based on a combination of the knowledge of Agile project management, Lean thinking, and collaboration research, which provides a holistic approach to assessing the functioning of workflows in practice. This holistic view is indicative of the complexity of modern SaaS organizations where development processes entail both technical and organizational as well as communication-based factors.

2.14 Summary of the Theoretical Framework

In this chapter, the main literature on the topic of project management, Agile methodology, Lean thinking, Kaizen, Value Stream Mapping, cross-functional collaboration has been reviewed. Combined, these views offer a theoretical framework to interpret the functioning of workflows in SaaS organizations and how to enhance them using an organized process analysis.

The literature points out that successful workflows in software development should not be reliant on technical processes only, but also effective communication and coordination among the teams and roles definition.

Agile methodologies offer flexibility in the development process, whereas the Lean way of thinking asks organizations to examine the workflow in a system-wide view. Kaizen focuses on constant improvement by making small changes until the process is completed, and Value Stream Mapping is also a visual tool that allows highlighting the inefficiencies in workflows.

The theoretical framework in this chapter, by combining these points of view, helps advance the goal of the study to analyze and improve the workflows in Company X. The framework not only offers conceptual guidance, but also offers analytical tools that are then used in subsequent chapters to analyze the process of development of the organization.

2.15 Conceptual Model for the Study

According to the literature reviewed, this thesis will use a conceptual model that combines Agile project management practices, Lean thinking, Kaizen philosophy and Value Stream Mapping to examine cross-functional workflows in a SaaS organization. Agile offers the operating environment where the development activity is structured with iterative cycles, regular feedback, and collaborative decision making.

Lean thinking brings about an emphasis on the enhancement of the flow of processes and reduction of inefficiencies in the whole system. Kaizen is the philosophy of improvement that promotes small and long-term process improvements. Value Stream Mapping is the analytical tool that bridges the gap between theory and practical work process, by visually illustrating the flow of work and information through the departments.

This is an integrated model that allows the study to assess performance of workflow without presupposing that the system is inherently defective. Rather, the framework acknowledges that even established workflows can have areas of potential improvement. The study determines specific improvements that can be made by examining the impact of communication, handoffs, and responsibilities on the workflow performance and, therefore, contributing to the long-term efficiency of Company X.

2.16 Linking Theory to the Research Methodology

The theoretical framework that has been formulated in this chapter is the direct informant to the methodological approach that will be employed in this study. Due to the aim of the research to gain insight into the functioning of workflows in practice, a qualitative case study would be suitable. This method enables the researcher to delve into the experiences and insights of the people who are part of the workflow as he/she studies the organizational environment in which the processes are taking place (Rother and Shook, 2003).

The primary method of data collection is semi-structured interviews since they will enable the participants to elaborate their experiences, observations, and perceptions of the workflow processes. Interviews will also be useful in terms of communication patterns, challenges in collaboration and informal practices which might be unrecorded in official documentation. Document analysis will be used to add to the information gathered in the interviews by analyzing internal workflow diagrams, ticket management systems, and process descriptions which explain how the work is supposed to flow within the organization.

Value Stream Mapping is a tool of connecting the gap between theory and practice because it translates qualitative information into a visual workflow representation. The study pinpoints delays, rework loops, or communication difficulties that arise by mapping the sequence of activities used to improve the products and resolve bugs. Kaizen is used to interpret the findings in such a way that the opportunities to improve are presented as small changes and not significant redesigns.

2.17 Relevance to Strategic Project Management

In terms of strategic project management, the workflow efficiency is strongly connected with the capacity of an organization to attain long-term goals. Strategic project management focuses on aligning activities in the project with the larger organizational objectives as well as ensuring that resources are utilized efficiently to generate value. Development workflows in SaaS organizations are a strategic capability as they determine the quality of products, the rate at which products are released and customer satisfaction (Womack and Jones, 2003).

Effective cross-functional teamwork of the Product, the Engineering, the QA and the Technical Support Engineering assists strategic goals through faster delivery of product improvement and faster response to the customer needs. On the other hand, the presence of inefficient workflows can cause delays, higher operational expenses and the lack of competitiveness in the fast-evolving software markets.

Kaizen and Value Stream Mapping are intertwined with a strategic project management framework that emphasizes the value of constant learning and improvement of processes. Companies that assess and refine their processes on a regular basis are in a better position to meet the changes in technology and the changing customer demands. Through the workflow analysis of the company x development, this thesis could be used to explain how the operational processes can be geared towards larger strategic objective within SaaS organizations.

2.18 Contribution of the Theoretical Framework

The theoretical framework designed in this chapter can be used to add to the current knowledge on the topic by illustrating how the ideas of Lean improvement can be modified to fit knowledge-intensive SaaS settings. Although the concepts of Kaizen and Value Stream Mapping are quite common in manufacturing, their usage in the context of software development workflow is underrepresented in the scholarly literature.

This study makes Lean principles applicable in digital workplaces by emphasizing the flow of information, instead of material flow. The combination of cross-functional collaboration, transparency of workflow and clarity of roles in the framework also offers a more holistic view on workflow performance. This combined method echoes the nature of the SaaS organizations whereby technical processes and human interactions are deeply interwoven.

The framework thus adds value both in theory and in practice through offering a systematic way of assessing work processes within the software development environment. It provides information about the ways in which organizations can use Lean-based improvement techniques to improve collaboration and ensure smooth development processes.

2.19 Chapter Summary and Transition

The chapter has synthesized and incorporated the literature on Agile project management, Lean thinking, Kaizen, Value Stream Mapping, and cross-functional collaboration. The combination of these concepts constitutes the theoretical basis of the analysis of the workflows in a SaaS development environment. The chapter outlines the significance of transparency, clarity of roles, flow of information and constant improvement in ensuring successful cross-functional cooperation.

The theoretical framework elaborated here is the foundation of the empirical research below. Combining the Kaizen and Value Stream Mapping with the knowledge of Agile and collaboration studies, the framework will allow performing a systematic examination of the processes between Product, Engineering, QA, and Technical Support Engineering in the Company X. The following chapter is the research methodology employed in this study which includes the qualitative case study design, data collection techniques, interview process, and data analytical process to study the working process in practice.

Chapter 3: Methodology

3.1 Introduction

This chapter explains the research process in this thesis. The purpose of the methodology section explains the manner in which the research was designed, data was collected, discussed, and how the research guarantees that it is valid, reliable, and ethical. A clearly definite methodology is significant as it enables the readers to comprehend as to how the research was conducted, conclusions were drawn and findings developed. By describing the research process in a transparent manner, the study ensures that the results can be evaluated and interpreted with confidence.

The objective of this thesis is to evaluate and enhance cross-functional workflows between Product, Engineering, Quality Assurance (QA), and Technical Support Engineering (TSE) within a Software-as-a-Service organization referred to in this thesis as Company X. These processes entail various teams collaborating in order to provide product enhancements and fix customer-reported bugs. Since the study is aimed at comprehending processes, collaboration and pattern of communication, as opposed to quantifying numerical indicators of performance, a qualitative research design is deemed the most suitable.

This chapter details the methodological decisions that inform the research. To begin with, the case study approach and research design are outlined. Second, the organisational background of the case study is presented. Third, data collection methods are detailed, such as semi-structured interviews, document analysis, and observation of workflow. Fourth, the analysis methodology that will be used is presented, with thematic analysis and Value Stream Mapping being mentioned. Last but not least, the chapter talks about reliability, validity, ethical issues and methodological limitations.

3.2 Research Design

This is a qualitative case study research study. The qualitative research approach is best applicable to the research that is oriented on the investigation of organizational processes, patterns of collaboration, and communication patterns. In comparison to quantitative research, which emphasizes the measurement and statistical association of numbers, qualitative research aims at comprehending how people experience and perceive organizational phenomena in their natural environment. This renders qualitative research particularly beneficial to investigate complex work processes that comprise several teams and developing practices.

Case study approach enables the researcher to study a phenomenon in a real life setting. Case studies are typical in organizational research when the demarcation between the phenomenon being researched and the environment are indistinct. The Product, Engineering, QA, and TSE processes at Company X are deeply integrated into the organizational structure, development tools and collaboration practices of the company in the current research, analyzing the workflows as a single case provides a comprehensive insight into how the process operates in real world and how it could be enhanced.

The study design may be referred to as exploratory as well as evaluative. The exploratory aspect involves understanding whether or not there are cross-functional workflows in the organization and identifying a pattern in communication and coordination across teams. The evaluative dimension deals with the assessment of the effectiveness of these workflows and how to determine how to enhance these workflows. Rather than testing the established hypotheses, the study will be analyzed with the aim of developing a practical understanding of the current workflow and commenting on the potential opportunities on how Kaizen and Value Stream Mapping can be used to maintain continuous improvement.

3.2.1 Justification for Qualitative Research Approach

Qualitative research is especially appropriate in the research that tends to comprehend multifaceted organizational functions and relations between human beings. The workflows in software development environments are usually influenced by formal procedures, but informal communication practices, collaboration patterns, and personal interpretations of duties. Quantitative research methods tend to be based on measurable performance measures like the number of years to develop, the number of defects, or the productivity measures Though these indicators will provide some valuable data on the performance outcomes, they will not be the causes that will determine the impacts of collaboration practices on the workflow efficiency.

The aim of this thesis is to learn how cross-functional teams organize their work and how communication practices can influence the effectiveness of the working process. These dimensions could not be easily quantified in terms of numbers because they involve subjective experiences, interpretation and considerations of contexts. The qualitative research methods allow the participants to elaborate their opinions and experiences in details such that the researcher can be in a position to indicate the patterns which cannot be identified with the help of quantitative researches.

The articulation of the other justification of the qualitative method is the exploratory nature of the study. This research is not aimed at testing a certain number of hypotheses that are predetermined but tries to understand how the current workflows operate and how they can be improved. Qualitative methods are particularly efficient when the researcher intends to carry out an exploratory study since it allows the researcher to investigate phenomena freely and flexibly. The respondents can use their own language to share their experiences and this may result in the creation of knowledge that may have not been captured by the structured questionnaires or the quantitative information.

In addition, the qualitative research assists in the workflow analysis as the socio-technical system. Such systems are very complex in their interactions between technology, processes and human behavior. To illustrate, the effectiveness of a ticket management system will not merely depend on the software tool itself but also the manner in which the teams will use the tool to communicate information. In order to understand these kinds of interactions, an approach is required that is a blend of both the technical processes and human experiences.

This study will be assisted by the qualitative research approach in exploring the interplay of collaboration practices, communication patterns and workflow structures at Company X. This approach is a contextual and rich picture of the development process that is needed to identify viable areas of improvement.

3.2.2 Justification for Case Study Methodology

The popularity of the case study approach in the study of organizational and project management is explained by the possibility to study the complex phenomena in their reality. Case studies can be particularly effective in cases where researchers want to know how processes work within specific organizational set-ups. Facilitating the isolation of the phenomenon under study and the context in which it is executed is often impossible in such instances because the processes under study are situated in the context of organizational structures, tools and practices.

In the current study, the Product, Engineering, QA and Technical Support Engineering workflows are directly related to the organizational environment of Company X. The factors that affect these workflows include the development tools used by the company, channels of communication, team structures and product management practices. Learning about these workflows out of the context of their organization would restrict the potential to learn how these workflows work in real life. Case study approach enables the researcher to study these interactions as a whole.

The other benefit of case study research is that it provides a chance to put in perspective several perspectives in one organization. Under cross- functional workflows different teams are likely to have different perceptions regarding processes. The engineers can be worried about the technical feasibility and system architecture, product managers may be worried about customer requirements and feature prioritization, and the QA specialists may be worried about risk mitigation and defects. The combination of these views analysis will provide a more accurate idea of the workflow.

Multiple data sources are also supported by the case study approach. A combination of interviews, documents and workflow observations may be created to form a comprehensive image of the process being investigated. This data triangulation enhances the validity of the research results since the researcher is able to confirm the observations using a variety of evidences.

It is due to these reasons that case study approach is a good fit to study cross-functional workflow in Company X. It helps the researcher to examine how the processes of development work in practice and at the same time takes into account the organizational environment under which these processes work.

3.3 Case Study Context

The case structure that was investigated in this paper is known as Company X to preserve the confidentiality and to save the identity of the organization. Company X is a Software-as-a-Service company that offers business management solutions to companies in the wellness and fitness industry. The platform incorporates a number of features such as scheduling, billing, customer relationship management, marketing automation, and reporting services. The system is used by clients to operate business daily and communicate with their customers.

The company is in a fast development cycle where new product enhancements and bug fixes are being done on a continuous basis. The development teams operate in cycles and the releases are also periodically delivered. Consequently, effective coordination among the departments is necessary to sustain quality on the products and also to deliver solutions to the customers at the right time.

The case study is centered around the interaction between four major functions that are entailed in the product development and support workflows: Product, Engineering, Quality Assurance (QA), and Technical Support Engineering (TSE). These departments work closely together in case of features improvement and customer-reports. The product teams set requirements and focus on improvements depending on customer needs. Technical solutions are applied in engineering teams. The QA teams ensure that the implemented features are of quality and functioning. Technical Support Engineering departments verify the issues reported by the customers, and give other technical information of the support point of view.

The researcher in this study works in Company X in a Product Operations position. This role offers firsthand exposure to the internal business operations, team work, and the development tools employed in the company. The ability of the researcher to access internal documentation, and workflow systems facilitates a comprehensive study of the processes which may otherwise be hard to observe. Simultaneously, this dual role has to be approached with much consideration to objectivity and ethical issues in order to make the research sound academically.

3.4 Data Collection Methods

In this research, the methodological triangulation of data collection is used i.e., multiple sources of qualitative data are required to increase the level of credibility and strength of the findings. Using different sources of data, when the researcher utilizes more than one source of data, will allow him or her to make comparisons across perspectives and to cross-check observations. Semi-structured interviews, study of internal documents, and workflow observation are the primary data collection techniques in this study.

3.4.1 Semi-Structured Interviews

In this study, semi-structured interviews will be used as a prime source of primary data. Such a type of interview will enable the researcher to address predetermined issues and also give the participants the freedom to elaborate more on their experiences and outlooks. Semi-structured interviews are better than highly structured interviews by encouraging open discussions and enabling the participants to give information that might not have been the foreseen results when the study was designed.

The main stakeholders who represent various positions in the development process are interviewed. The interview sample comprises of three primary stakeholder perspectives: Product, Engineering/QA and Technical Support Engineering. These roles were chosen as they are the various points in the workflow and offer complementary information on how tasks can be passed through the initial request to the final release.

The interviews will take about 30-45 minutes each and will be conducted in a one-on-one manner to enable the participants to express themselves freely about their experiences. The interviews are recorded with audio and video recording with prior consent to allow the accuracy in analysis. Once the interviews are conducted, the recordings are analyzed and transcripted to establish the main themes and patterns connected with the workflow efficiency and collaboration practices.

An interview guide is employed to provide uniformity in interviews. The guide has a series of fundamental questions pertaining to workflow communication, role duties, collaboration issues, and enhancement possibilities. There are also some role-specific follow-up questions to investigate team-specific issues. This design will make the interviews comparable and at the same time give the participants the opportunity to give in-depth information about their respective positions.

3.4.2 Document Analysis

Interpretation of documents is employed in order to supplement interview information and to give an understanding of processes that are stated in an organization. The areas of internal documentation analyzed in the study are workflow descriptions, process diagrams, issue tracking systems and collaboration platforms utilized by development teams. These papers are the formal framework of procedures and a point of reference about the way work procedures are expected to work.

The analysis of such documents assists in determining the major workflow steps and points of decision and handoff between team members. As an example, the issue tracking systems may show the flow of feature requests and bug reports through the various workflow states. Documentation platforms give visibility into the way requirements, technical specifications and testing procedures are documented and communicated.

The document analysis is also useful in determining any possible discrepancies between the formally defined processes and real practices as narrated by the participants of the interview. Such comparison enables the researcher to investigate the functioning of the workflows in practice and the presence of either informal adjustments or workarounds in the system.

3.4.3 Workflow Observation

Workflow observation entails the analysis of movement of tasks and ticket through development phases with collaboration and project management tools. Observation is made on the transitions of the work between Product, Engineering, QA, and Technical Support Engineering and the communication between them.

The activity of the workflow is also observed to gain insight into the practical aspects of coordination, like how the update of tickets is done, how the team communicates during the handoff, and how delays or loops of rework can happen. Such observations can be used to determine trends that might not be clearly written down in formal process descriptions.

In the observation of workflow, it is more effective in determining inefficiencies in regards to waiting times, role ambiguity or redundancy. The combination of observational data and the interview responses, together with the analysis of the documents, helps the study to create an in-depth picture of the functioning of the workflows in Company X.

3.4.4 Selection of Interview Participants

The choice of the participants of the interview is also significant to make sure that the research would be able to obtain a complete picture of the workflow that is being studied. In qualitative studies, purposive sampling is the most common method of selecting participants, i.e. the participant is chosen due to their expertise and/or participation in the phenomenon under investigation. This will make sure that the participants are relevant and can contribute insight into the topic of the research (Yin, 2018).

The participants of the interviews in this research were chosen among the teams that are directly involved in the product development process of Company X. The process workflow explored in the current study includes multiple phases, as it starts with receiving feature requests and bug reports and concludes with the publication of product enhancements. Several teams are involved in this process and, that is why, it was essential to involve the representatives of various functional areas.

The participants were thus identified in Product management, Engineering and Quality Assurance and Technical Support Engineering. These roles are the outlined milestones of the working process and offer the complementary points of view regarding how the tasks are flowing along the development process. Product managers are responsible for defining requirements and prioritizing improvements based on customer needs. Engineers implement the technical solution, while QA specialists verify that the implemented functionality meets defined quality standards. Technical Support Engineering teams evaluate customer-reported issues and provide additional technical context when necessary.

By interviewing participants from these different roles, the study captures a cross-functional view of the workflow. This approach helps identify how communication occurs between departments and where coordination challenges may arise. It also allows the researcher to compare perspectives from different stages of the workflow and identify areas where expectations or interpretations differ.

3.5 Data Analysis Approach

In this study, data analysis will be based on thematic analysis strategy with process-oriented analysis by means of Value Stream Mapping. Thematic analysis is the process that results in finding common themes and trends within qualitative data (Braun and Clarke, 2006). Summaries of interviews are assessed and put into thematic groups based on communication habits, collaboration issues, role clarity, workflow clarity, and areas of improvement.

Value Stream Mapping is an analytical tool employed to illustrate the present position of workflow, as well as to create better future-state workflow. Data collected during interviews, documents, and observations of workflows are mapped on the workflow stages to draw value-adding and non-value-adding activities. The waiting times, rework loops and communication delays are looked into so as to have an insight into how the workflow can be improved.

Kaizen principles are used to interpret the findings by focusing on improvement in small steps as opposed to redesigning a process on a big scale. This method also supports the objective of finding efficient and sustainable solutions that reinvent current workflow without causing any instability in the organization.

3.6 Reliability and Validity

Another important aspect of qualitative research is its reliability and validity. The notion of reliability is linked with the uniformity of the research procedure, and that of validity with precision and trustworthiness of the findings. This study maintained reliability by making sure that there was a uniform interview guide, data collection processes were well documented and detailed notes were available by the researcher.

Triangulation is a methodological way to increase validity. The research involves the use of a mixture of interviews, document analysis, and workflow observation in order to compare different perspectives and verify the findings in a wide range of data sources. This will reduce the risks of relying on a certain view and will increase the legitimacy of the results.

The insider position of the researcher in Company X makes him/her have good contextual knowledge but it introduces a potential bias. The purpose of the study is to analyze processes rather than to measure the performance of single individuals in order to eradicate this risk. The data sources are compared with different sources, which ensures that the outcomes are grounded on the workflow system, rather than personal interpretations.

3.6.1 Triangulation of Data Sources

One of the main approaches of qualitative research is triangulation that increases the credibility and strength of the results. The concept of triangulation implies the utilization of a variety of data sources, methods or perspectives to study the same phenomenon. When the results of various sources are comparable, the researchers may be more certain about the correctness of the results.

The triangulation in this research is attained through the use of three major data sources, including interviews, document analysis, and workflow observation. All these sources give us a distinct view of the development workflow in the Company X. Interviews will give us the experiences and interpretation of the workflow itself as viewed by the people who were directly involved in the workflow. The analysis of documents gives an idea concerning the formally defined processes and organizational guidelines. Observation of the workflow shows the flow of tasks within the system in practice.

When these sources are compared, the researcher can determine whether there are any similarities and differences between official processes and the real practices. As an example, a workflow can be described in internal documentation in a particular way, and the participants of the interview can describe modifications or informal practices that can happen in the context of daily operations. Monitors the workflow of project management systems are used to check how work really flows in the system.

This triangulation method will enhance the validity of the research study since there are chances that such a method will minimize the chances of using one source of information. Rather, the work analyses the workflow by looking at it in various points of view to create a holistic understanding of the workflow.

3.7 Ethical Considerations

Ethics will be a factor in this study. They are part of the interviews on a voluntary basis and the participants are made aware of the aim of the study before the commencement of the data gathering. The interviews are recorded with consent provided beforehand and the participants are aware of the use of the data in the research.

To safeguard the identity of the company and the employees, all of the information in the organization utilized in the thesis is anonymized. Because of this fact, the company will be called Company X in the thesis. The reports of interviews are not based on the statements of particular people but rather on their functional level.

The information gathered in the course of the research study is academic only and is addressed in the context of university research ethics. The thesis does not contain confidential information that would identify individuals or any sensitive information about the organization.

3.8 Limitations of the Methodology

Despite the fact that the qualitative case study approach is very insightful on the workflow of an organization, it has its limitations as well. The research is based on one organization, and therefore, the results might not be directly applicable to other SaaS companies. The aim of a research is, however, analytical generalization as opposed to statistical generalization. The lessons learned in the case study can be applicable to organizations that have a similar structure and development process.

The other weakness is associated with the dual role of the researcher as an employee and a researcher. Although this role offers a good access to the internal processes and situational insight, special focus must be paid to objectivity. Transparency in data collection and analysis will help overcome this challenge.

These limitations notwithstanding, the ensemble of several data sets and a logical analysis gives a solid ground to assess the workflow in Company X.

3.9 Chapter Summary

In this chapter, the research methodology of the study has been outlined. It described the qualitative case study design, the setting of the case organization, and data gathering techniques applied to study the cross-functional workflows. It was conducted in form of semi-structured interviews, document analysis, and observation of workflow to gain a deep insight into the flow of tasks between Product, Engineering, QA and Technical Support Engineering teams.

The chapter also outlined the method of data analysis grounded on thematic analysis and Value Stream Mapping and the steps taken to guarantee reliability, validity, and ethical considerations. The approach described below is the basis of the empirical analysis in the next chapter. The following chapter includes the results of the research and considers the existing workflow through the Value Stream Mapping and feedbacks provided by the participants of the interview.

Chapter 4: Empirical Findings

4.1 Introduction

This chapter summarizes the empirical results of the research written on the basis of the interviews with the main stakeholders of Company X and the analysis of the workflow documentation and Value Stream Mapping (VSM) exercises. This chapter is aimed at gaining the in-depth insight into the way feature improvement requests travel within the organization and how the interaction between teams impacts the efficiency of the entire development process.

The empirical study is based on the interaction of Product, Engineering, Quality Assurance (QA), and Technical Support Engineering (TSE) since these departments are united to control the lifecycle of product enhancements and features requests. It is significant to understand the interaction between these teams since the software development processes in SaaS organizations rely on interaction and coordination between various functional roles (Dingsøyr et al., 2012).

The qualitative data serves as the foundation of the empirical results reported in this chapter because the authors use semi-structured interviews as a method of gathering the data on three stakeholders who represent major functional areas of the workflow. All interviews gave a new orientation of the process, and the researcher managed to learn how the workflow takes place at various levels of development.

The participants of the interviews explained the manner in which requests are received, validated, prioritized, implemented, tested, and finally put into production. Such views assisted in determining the area of delays and areas of improvement.

Besides interviews, internal documentation and process diagrams were studied to get an insight into the formally defined workflow that the organization uses. These were documents which contained internal workflow descriptions, issue tracking structure and process diagrams which outline the flow of work within the system.

The analysis of documents contributed to the possibility of official representation of the process that could be compared to the experiences that were described by the interview participants. This type of comparison is also typical in qualitative case studies where it helps to demonstrate the discrepancies between formal process and actual practice in the organizations (Rother and Shook, 2003).

One of the key elements of the empirical research was the development of Value Stream Maps with the help of the internal stakeholders. The whole workflow process, starting with the first feature request and ending with the final release was visualized using Value Stream Mapping. The VSM activity enabled the research team to determine the waiting times, work-in-progress, and processing times at various workflow stages. The metrics have allowed assessing the efficiency with which the system is used and the locations of non-value-adding processes. VSM has frequently been applied in the study of Lean management to uncover latent inefficiencies in the conventional workflow models (Womack and Jones, 2003).

The empirical results are presented in a number of sections. First, the chapter outlines the existing workflow of Company X to process feature improvements. Second, areas of inefficiencies found in the existing workflow are outlined and discussed. Thirdly, the findings of the Value Stream Mapping exercise are presented, such as the determination of delays and non-value-adding activities. Lastly, the chapter outlines the design of a better future state workflow and assesses the possible impact of the suggested improvements.

With the help of the thematic analysis, the knowledge gained during the interviews was put together into the major patterns that were shared among different stakeholder positions. Table X consolidates the key results of the interviews, stakeholder attitude, and the issues identified and how they can be used to improve work.

Table 4.1: Interview Analysis of Stakeholders

Stakeholder	Role / Department	Analytical Perspective	Key Insights	Identified Issues	Contribution to Improvements
Stakeholder 1	Technical Product Manager	Thematic analysis (requirements & prioritization)	Emphasized importance of clear requirements and customer value alignment; backlog contains many unrefined items	Large backlog size, unclear requirements, delayed prioritization	Suggested backlog triage meetings, improved requirement clarity, early collaboration with Engineering
Stakeholder 2	VP of Engineering	Process and workflow analysis (development & testing stages)	Highlighted delays due to unclear specifications and batch testing; context switching affects efficiency	Rework due to unclear requirements, testing bottlenecks, batch processing delays	Recommended smaller work batches, continuous testing, early technical involvement in design stage
Stakeholder 3	Manager, Technical Support Engineering	Validation and intake analysis (issue identification & filtering)	Focused on importance of validating requests before Product stage; many requests lack sufficient detail	Invalid or unclear tickets, repeated clarification cycles with Product	Proposed stronger ticket validation, better communication with Support, improved initial data quality

The table demonstrates that, despite the fact that each stakeholder works in a different part of the workflow, there were challenges that were shared by all roles that were similar, particularly those related to communication gaps, wait times, and ambiguous requirements. The Value Stream Mapping analysis and the future state workflow design both benefited from these findings.

4.2 Overview of the Current Workflow (AS-IS Process)

The current process of managing feature improvements in Company X will start with the reception of a request to create a new feature or improvement by a customer. These requests are usually expressed by the customers via the Support team as the main point of contact between the organization and the users. The support agents gather the appropriate data regarding the request and prepare an internal ticket with the offered improvement.

This ticket typically has the description of the problem made by the customer, the situation, within which the problem is observed, and other details, which could assist internal teams to comprehend the request. Software development environments also tend to use issue tracking systems such as Jira to record and manage these requests as it provides visibility and traceability across teams (Sommerville, 2016).

After creating a ticket, it is sent to the Technical Support Engineering team. The TSE team conducts a preliminary technical check of the request. This step of validation, according to the interview participants, is required due to the fact that not all customer requests can be considered a real product improvement. Very often, the requests are founded on the misinterpretation of the current functionality or even have no technical information needed to be implemented.

The TSE team role is thus to interpret the request, confirming whether it represents an actual product gap and to make sure that the ticket has adequate information and then it is forwarded to the Product team. This is done to minimize redundant work on Product and Engineering teams and also enhances the quality of the incoming requests.

Once TSE validates the request, it passes on to the Product team where it is assessed and prioritized. Product Managers consider the request and decide whether it can fit in the overall product strategy and roadmap. This assessment can be done by examining the customer responses, examining comparable requests by other clients, and estimating the value that the enhancement will provide to the users.

In other instances, Product Managers can perform user interviews or collect more information about customer to help clarify requirements and understand the desired functionality. These actions are typical of Agile product development settings, in which customer feedback is significant in product development (Beck et al., 2001).

In case the request is deemed valuable and viable, it is included in the product backlog. The backlog is a list of possible improvements and development activities, which are not yet scheduled to be implemented. The backlog in most software organizations is continuously increased as new requests are made. The backlog offers a formal manner of handling the work that is received, but it may cause delays when the rate of requests increases beyond the capacity to develop the organization (Poppendieck and Poppendiek, 2003).

The feature requests can be kept idle during the backlog stage before they are picked to be developed. Interviews held as part of the research showed that the backlog at Company X has a high amount of feature requests that require prioritization. Such a huge backlog would lead to a list of tasks to be examined and prioritized by the Product team before they would be allowed to proceed with the process. These types of queues are usually identified with waiting time in the Lean process analysis and are regarded as a type of process waste as they cause a delay in delivering a value to a customer (Hines et al., 2004).

When a feature request is chosen to be developed, the Product team starts specifying the requirements in greater detail. This phase usually involves the creation of user stories, creation of acceptance criteria, and creation of functional specifications (Cohn, 2004). The design team can also produce interface mockups or visual prototypes which can depict how the new functionality would look in the product interface. These design artifacts assist the developers to know the desired user experience and minimize chances of being misinterpreted when implementing.

Following the design stage, the request moves to the estimation and planning phase. During this stage, Engineering teams evaluate the technical complexity of the feature and estimate the effort required to implement it. The estimation is usually done in story points which are not in real development time but in relative effort. Story points enable teams to rank tasks by their complexity and workload expectations instead of trying to determine exact times. Such an approach is common in Agile software development as it promotes collaborative estimation and minimizes the stress to create precise time predictions (Schwaber and Sutherland, 2020).

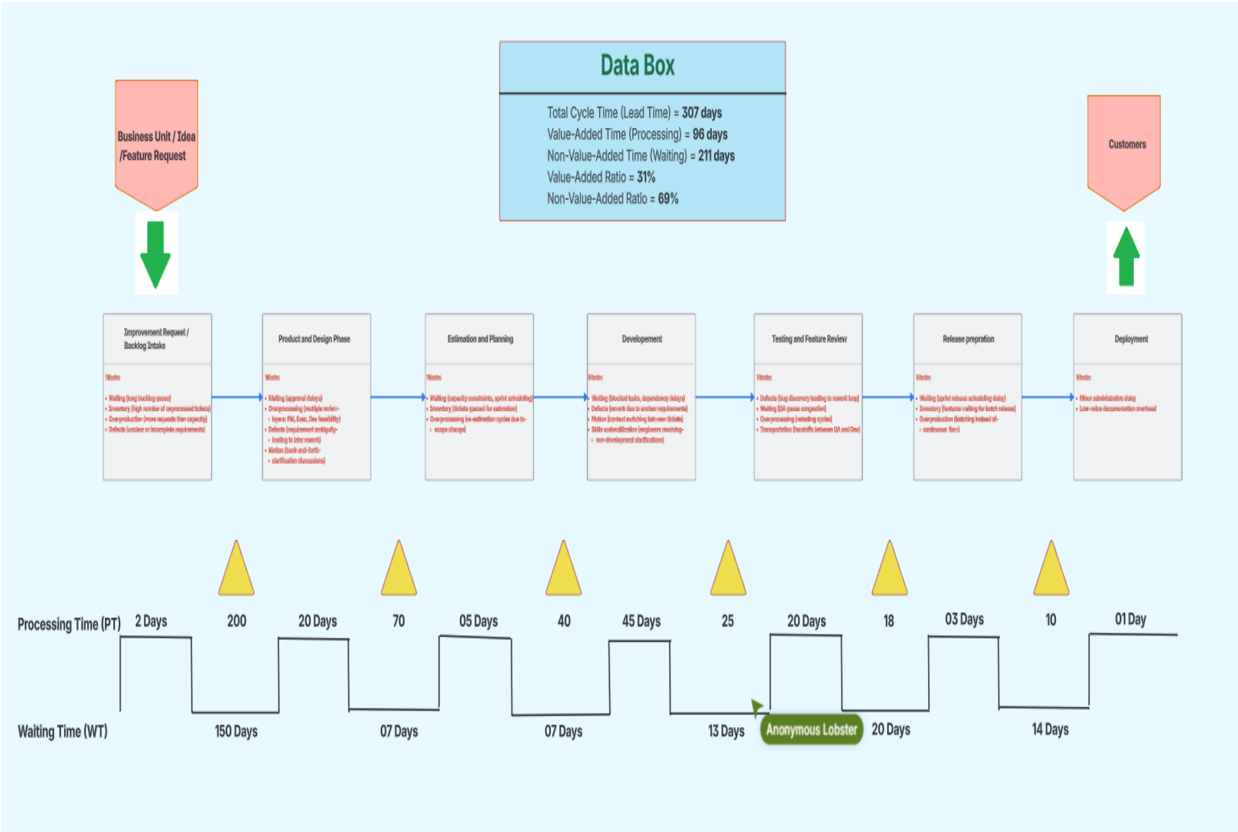
After estimation, the request is put forward to be developed. The developers start working on the implementation of the feature based on the requirements and design specifications. The development work can be the modification of the existing system parts, the development of new modules, or the addition of new functionality to the existing architecture. Once development work is done, the feature is handed over to the Quality Assurance team where it is tested and validated.

QA team does functional testing to ensure that the implemented feature works as per the requirements. Regression testing is also undertaken by QA specialists to make sure that the new functionality is not causing unwanted problems elsewhere in the system. When defects or inconsistency is found out in the testing, the feature would be sent back to the Engineering team to rectify it. This cycle can repeat several times until the feature is of quality enough to meet the quality requirements.

After successful testing, the feature is finally validated by the Product team. Product Managers test the functionality done in a staging environment to ensure that it is consistent with the initial design and satisfies customer requirements. Once this validation is complete, the feature will be released and deployed to the production environment as per the release schedule of the organization.

Despite the fact that this workflow structure offers a well defined series of tasks to be performed by each team, empirical analysis showed that most of the total lead time is not related to actual work in development but is in waiting between tasks. The waiting time between jobs is regarded as one of the major types of wastes in the Lean management as it does not allow work to be easily run through the system (Ohno, 1988). These inefficiencies are discussed in the sections below based on knowledge gained through the interviews and analysis of Value Stream Mapping.

Figure 4.1: AS IS Value Stream Mapping



4.3 Identified Inefficiencies in the Current Workflow

The empirical analysis of the current workflow revealed that the primary inefficiencies in the system were not related to the technical complexity of development tasks but rather to the coordination and waiting times that occur between different stages of the process. Stakeholder interviews were done with Product, Engineering and Technical Support Engineering, and it was consistent that most delays were noted when tasks were awaiting a review, prioritization or assignment and not the actual implementation. This finding is consistent with earlier studies on Lean software development, which demonstrate that information flow latencies can be a bigger contributor to the total lead time than the actual development work (Anderson, 2010).

A major inefficiency revealed during the analysis was the backlog of tasks in the product backlog. The backlog is likely to increase with time due to the constant inflow of new feature requests by customers. As much as the backlog offers a well-structured mechanism of storing the possible improvements, it also forms a queue where requests stay idle over a long period. In the Value Stream

Mapping activity, which was undertaken in this paper, the backlog phase had about 200 work items that were awaiting to be reviewed and prioritized. Although the processing time it needed to assess a request was not very high, the waiting time to be reviewed was much higher. This kind of delay is defined as waiting waste in the Lean terminology wherein work is not deployed to contribute value to the end product or service (Reinertsen, 2009).

Another inefficiency observed in the workflow was related to the design and specification stage. During this phase, multiple design tasks are often handled simultaneously. While parallel work can increase productivity in some contexts, it may also lead to coordination challenges when teams must manage many tasks at the same time. Interview participants indicated that the Product and Design teams frequently manage dozens of feature specifications simultaneously, which increases the cognitive workload of team members and can slow down decision making. In the Value Stream Mapping analysis, this stage was associated with approximately 70 work items being processed in parallel. Large numbers of concurrent tasks often reduce overall workflow efficiency because they increase context switching and reduce focus on individual tasks (Liker, 2004).

The estimation stage also contributed to delays because estimation activities were typically organized in scheduled planning sessions rather than performed continuously. As a result, tasks that were ready for estimation sometimes had to wait until the next planning meeting before they could proceed to development. Although the estimation activity itself required relatively little time, the scheduling of estimation sessions introduced additional waiting periods within the workflow. This is typical of the batch based planning environment where the work is in cycles and not a continuous flow in the system (Reinertsen, 2009).

There was generally efficiency in the development activities in the Engineering team but inefficiencies were still witnessed because of context switching. Developers might simultaneously work on several features, especially when several tickets are allocated to a development sprint. Although this practice has the potential to enhance better use of resources, it can also decrease efficiency in the workflow by raising the amount of time spent to accomplish each task. Studies conducted on Lean software development indicate that productivity can be enhanced through limiting the work in progress by enabling the teams to concentrate on a fewer number of tasks simultaneously (Poppendieck and Poppendiek, 2003).

The test phase also posed a number of workflow issues. The QA team often has to work on several completed features at a given time, especially when development work is completed towards the end of a sprint cycle. This creates a testing bottleneck, with certain features held waiting until the capacity to do testing has become available. These bottlenecks are common in systems where the upstream work is done in batches and not spread evenly over time. When testing capacity is limited, work done is held up and postpones the flow of work towards release (Humble and Farley, 2011).

Another area that had delays was release preparation. The finished features are often bundled and released during release cycles in the present working process. Although release batching can be an easier method of coordinating deployment, it can also cause higher waiting times of completed features that are ready to be released but have to wait until the next scheduled deployment period. This form of delay is classified as overproduction or batching waste in Lean process analysis since the work is accomplished before it is required but cannot be provided at the moment (Kim et al., 2016).

Besides these operational inefficiencies it was also observed in the course of the analysis that there were communication related inefficiencies among the teams. Interview reports revealed that there was occasionally a necessity to repeat information several times among Product, Engineering, and QA teams.

To illustrate, in cases where the acceptance criteria were not elaborated, QA specialists would seek clarification on the expected functionality. Otherwise, in some cases, developers would need further clarification of design specifications before they could be implemented. Such circumstances caused the rework cycles that prolonged the duration needed to accomplish some tasks. In Lean terminology, rework is classified as defect waste since it is work that should be repeated either as a result of incomplete or wrong information (Forsgren et al., 2018).

In general, the workflow inefficiencies revealed during the current workflow were mainly associated with coordination, communication, and process organization, but not technical issues. These results indicate that lead time can be significantly improved by working on workflow design and communication practices without having to develop some extra resources.

4.4 Value Stream Mapping Results

The Value Stream Mapping activity ensured a comprehensive visual representation of the movement of feature requests through the Company X development process, by mapping both the stages involved and the processing and waiting times of the process, the VSM analysis allowed quantifying the effectiveness of the current workflow. Value stream Mapping is a popular tool of Lean management to analyze the system-level performance and opportunities of the complex workflow improvement (Rother and Shook, 2003).

The analysis revealed that the total lead time required to deliver a feature from initial request to final release was approximately 307 days in the current workflow. Lead time refers to the total duration required for a task to move through the entire system, including both active processing time and waiting time between stages. Within this total lead time, only 96 days were associated with value-added activities, meaning that the majority of the process duration was spent waiting between stages rather than actively producing value for the customer.

The difference between total lead time and value-added time represents non-value-adding activities within the workflow. In the current process, non-value-adding time accounted for approximately 211 days of the total workflow duration. These delays were primarily associated with backlog waiting, scheduling delays, coordination gaps between teams, and batch-based work practices. According to Lean process theory, identifying and reducing non-value-adding time is one of the most effective ways to improve system efficiency without increasing workload or resources (Womack & Jones, 2003).

Process efficiency can be calculated by dividing value-added time by total lead time. In the current state workflow, process efficiency was approximately 31 percent. This means that less than one third of the total time required to deliver a feature contributed directly to creating value for customers. The remaining time represented waiting, coordination delays, and other forms of process waste.

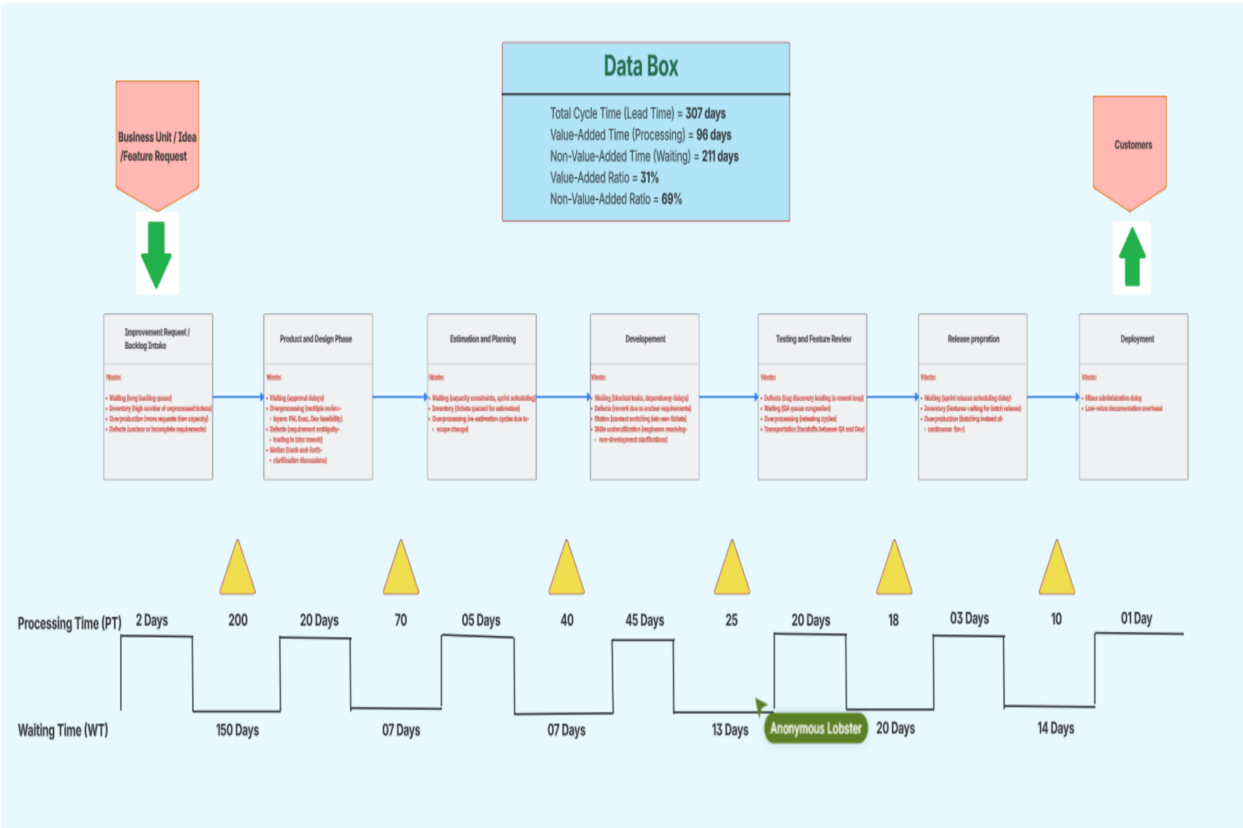
Although the efficiency level observed in the current workflow is not unusual for complex software development environments, it indicates that significant improvement potential exists. Many Lean studies have shown that knowledge-intensive workflows often contain large amounts of hidden waiting time that can be reduced through improved coordination and workflow design (Hines et al., 2004).

The Value Stream Mapping exercise also revealed that delays were not evenly distributed across the workflow. Most of the waiting time was spent in the backlog phase and during the development phases as opposed to the active work phases like code writing and testing. This observation indicates that the enhancement that aims at minimizing queues and enhancing the smooth flow of work may result in a substantial decrease in total lead time.

The next valuable lesson of the VSM analysis was that there are large work-in-progress queues at various workflow stages. Extremely high WIP was observed in the backlog stage and design stage, where there were a significant number of tasks waiting to be handled. The Lean process management regards excessive work in progress as a key contributor to inefficiency due to the fact that it causes more waiting time and poor responsiveness of the system (Reinertsen, 2009).

The visual representation created by Value Stream Mapping allowed the stakeholders to comprehend how delays can add up throughout the workflow and how waiting times at each stage, even those of seemingly small magnitude, can add up to very long lead times. The VSM exercise revealed the existence of these delays, thus serving as a starting point when developing a more effective workflow organization.

Figure 4.2: AS IS Value Stream Mapping



4.5 Design of the Future State Workflow

After examining the existing workflow, a future state workflow was created in liaison with the stakeholders of Company X. The future state design was not aimed at redesigning the entire development process but to add some incremental enhancements that would decrease the delays and enhance communication among teams. This philosophy is in line with the Kaizen philosophy that promotes continuous and gradual improvement over major change in an organization (Imai, 1986).

Among the most essential enhancements that were implemented in the future state workflow, the introduction of the practice of backlog triage should be listed. Rather than letting feature requests pile up in a big backlog, the future workflow will introduce more frequent backlog review meetings, and constrain the amount of work in progress in the system. Work in Progress (WIP) limits are implemented to minimize queues and enable work to flow faster through the workflow. Studies on Lean management have always indicated that, minimization of the WIP levels enhances responsiveness in the system and minimization of the general lead time (Anderson, 2010).

The other enhancement that was brought in the future workflow was more work between Product and Engineering teams in the initial phase of defining the features. In the modern workflow, the design work can be started before the technical feasibility is discussed to the full extent in the technical community. This occasionally creates further loops of clarification further into the process. Future workflow will promote prior technical consultation between Product and Engineering teams and will help point out areas of challenge before an in-depth design work is done.

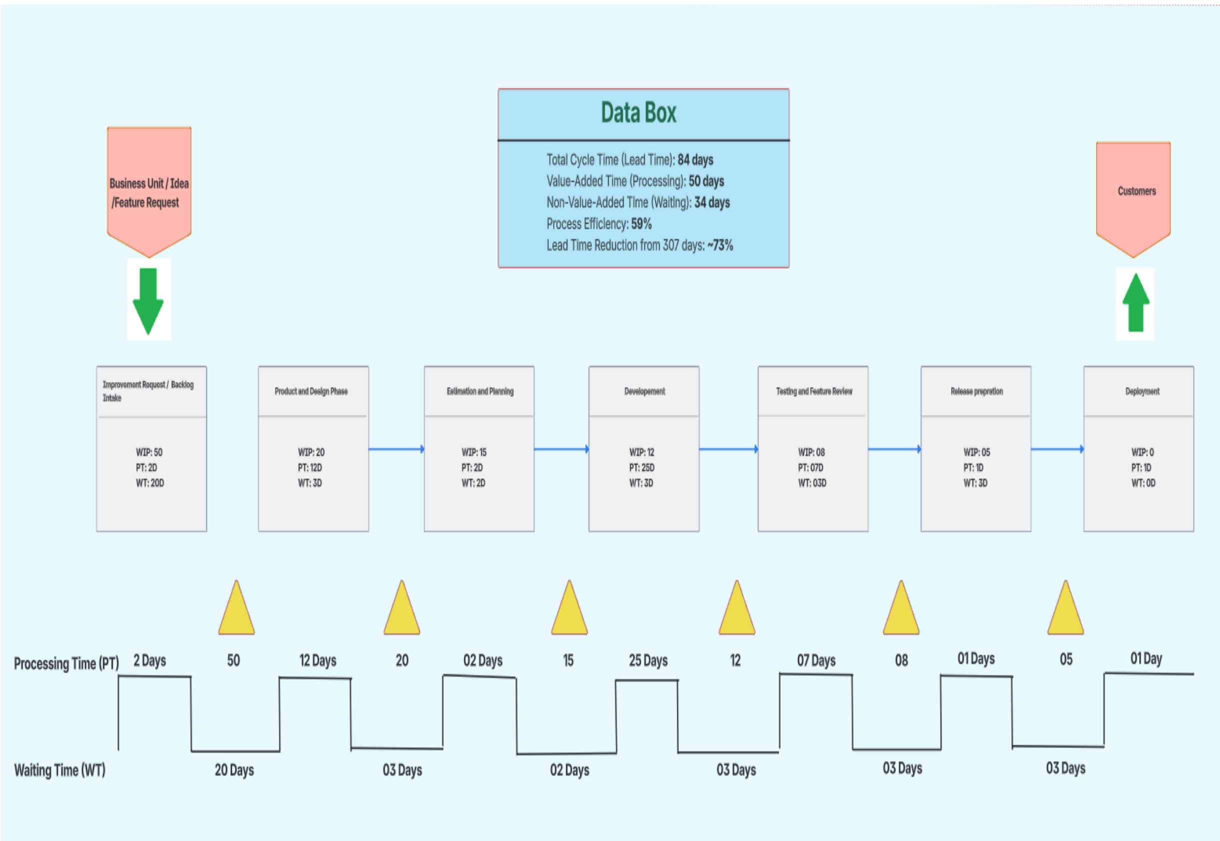
The estimation process was also redone to accommodate continuous estimation practices. The future workflow enables the Engineering teams to assess tasks when they are in the development pipeline rather than wait until it is time to have a scheduled estimation meeting. This method decreases the delays that go with batch-based estimation and enables tasks to flow in between stages with fewer delays.

The other enhancement is that smaller development batches are used. Large deliverables are broken down into smaller deliverables that can be deployed and tested faster. Smaller work items are typically completed quicker in the system and minimize the possibility of creating a bottleneck at the lower stages of the system like testing or release preparation.

It also enhanced testing practices with the introduction of the higher focus on automated testing and continuous integration practices. Automated testing enables the QA department to perform functionality checking faster and less manual work is needed to perform regression testing. The practice of continuous testing also aids in the earlier identification of problems during the development cycle thereby minimising the need to reiterate testing cycles towards the end of the development cycle.

Lastly, release management practices were enhanced through the introduction of more relaxed deployment mechanisms. Rather than having large release cycles, the future workflow enables smaller updates to be deployed more often. This can be done through technologies like feature flags and automated deployment pipelines that enable new functionality to be added slowly without having to do giant release coordination.

Figure 4.3: Future Value Stream Mapping



4.6 Impact of the Future State Improvements

The development of the future state workflow resulted in a number of significant changes in the efficiency of the entire development process at Company X. The biggest enhancement that was realized in the process of the Value Stream Mapping exercise was the total decrease in the lead time that was needed to deliver a feature through the first request stage to the ultimate stage of release. In the present condition of workflow in the state, the overall lead time was estimated at circa 307 days. A lead time of about 84 days was achieved after the suggested improvements were put in place. This is a cut off almost 73 percent on the duration taken to introduce a feature to customers. These reductions are usually witnessed in Lean improvement efforts where the main emphasis is put on the waiting time and workflow flow other than on the production capacity (Womack and Jones, 2003).

Most of this enhancement has been done through minimizing non-value adding processes in the workflow. In the present state process, the waiting time between stages, backlog buildup, scheduling delays and batch based work practices took on a total of 211 days of the overall lead time. The future state workflow dramatically helped to cut these delays by introducing backlog triage meetings, work in progress, and team to team communication. In the new process design, the non-value-adding time was decreased to about 34 days. This decrease shows how coordination and workflow organization can result in big performance improvements without necessitating new development resources.

The other significant change that was experienced in the future state workflow was the enhancement of the processes. The actual efficiency of the process is computed by dividing value added time by total lead time. Process efficiency in the present state workflow was about 31 percent, i.e., less than one third of the total workflow was used to create actual value to customers. The process efficiency went up to around 59 percent after the suggested improvements. This value, despite the fact that it is still an indicator of the existence of some waiting time in the system, is a major improvement to the original process structure. One of the main goals of Lean process improvement is to increase process efficiency to make sure that more productive activities are performed instead of wasting time due to their coordination (Womack & Jones, 2003).

Other improvements in communication and collaboration among teams were also introduced by the future state workflow. Small work items can be estimated, implemented and tested easily and this minimizes the chances of delays and coordination problems. Interview participants indicated that earlier collaboration between Product and Engineering teams during the requirement definition stage helped reduce misunderstandings later in the development process. When technical feasibility discussions occur earlier, developers are able to provide feedback on design decisions before implementation begins. This reduces the likelihood of rework and minimizes the need for clarification during later stages of the workflow.

There were also improvements in testing activities. The QA team can test new features faster and more consistently by incorporating more continuous testing practices and use of automated testing tools. Automated testing minimizes the amount of manual effort involved in regression testing, and enables defects to be identified earlier in the development cycle. Early defect identification has been generally accepted as one of the reasons that enhance the efficiency of software development since problems can be resolved before it spreads to subsequent portions of the workflow (Humble and Farley, 2011).

Another benefit of the future state workflow is the reduction of batch-based work practices. In the original process, tasks often moved through the system in large groups, particularly during estimation, testing, and release preparation stages. Batch processing can create bottlenecks because downstream teams must process many tasks simultaneously. By introducing smaller work batches and more continuous flow practices, the future workflow allows tasks to move through the system more smoothly.

On balance, the future state workflow shows that the development of a program can be enhanced by small steps towards Lean and Kaizen principles. Instead of making disruptive changes in the organization, the improvements are aimed at streamlining the current processes and enhancing the interaction of teams with each other. This is a good fit with the philosophy of continuous improvement that is extensively popular in Lean management literature (Liker, 2004).

4.7 Lean Waste Classification in the Current Workflow

A significant component of the Value Stream Mapping analysis was to determine various forms of waste that existed in the actual working process. Waste in Lean management is any activity that burns resources and does not add value to the customer. The idea of waste, which was initially applied in factories, has been transferred to knowledge-based work like software development. Even though the type of waste could vary in digital processes, the general concept is that any process that slows value addition or creates wasteful work needs to be on the list of improvements (Ohno, 1988).

Waiting waste was one of the most noticeable types of waste witnessed within the workflow of Company X. Waiting was experienced when workflow was idle as teams awaited reviews, approvals or availability of resources. The backlog phase was the most important area of waiting waste, with feature requests sometimes waiting a long time before being looked at and prioritized. Waiting waste was also encountered between the development and tests phases whereby features that were completed had to wait until QA capacity.

Overproduction was another category of waste that was detected in the workflow. Overproduction in the context of software development may take place when work has been done sooner than it is needed or when work is developed in excess of what is needed. This was the type of waste that was witnessed in the preparation of releases, whereby completed features were not active until they were deployed at specific times. Though the development work was already done, it was not possible to deliver the functionality to the customers until the next release cycle.

The waste of overprocessing existed in several workflow stages as well. Overprocessing is the act of undertaking more steps or activities which do not add value to the customer. Indicatively, frequent clarification cycles across teams occasionally necessitated more meetings or updates in documentation. Although coordination requires communication, a repetition of clarifications may mean that the information was not that clear in previous phases of the process.

The defect waste was discovered in the case when features were to be sent back to the Engineering team during the QA testing because of the lack of the functionality or the wrong implementation. Defect correction is an inherent aspect of the software development process, but too much rework can elevate the length of the development process, and lower the overall efficiency. Defect waste is a common indicator of poor specification of requirements or design specifications in earlier phases of the workflow.

Motion waste was another form of waste that was identified during the analysis, in knowledge work, motion waste is defined as a fromless moving of information across systems or teams. As an illustration, where there is need to seek more clarification between Product and Engineering teams, there can be information exchange several times before development can commence. These exchanges are repeated, making them to take more time to finish tasks and workflow delays.

Waste of inventory in the form of surplus work in progress was also found during the backlog-stage and design-stage of the workflow. Long waits are caused by high numbers of tasks in queue that make the development system less responsive. Restricting the amount of work in progress is thus a significant measure towards enhancing efficiency in the workflow.

The Value Stream Mapping exercise helped to recognize these types of waste, which offered a systematic mechanism of grasping the underlying factors behind the workflow inefficiencies. By tackling these types of waste by means of incremental improvements, the organization was able to dramatically decrease the lead time and enhance process flow.

4.8 Linking Empirical Findings to Kaizen and Lean Theory

The results of the empirical research provided in this chapter are highly corroborative of the theoretical notions in the thesis. Specifically, the findings demonstrate how the principles of Lean and the philosophy of Kaizen can be used to enhance the workflows in the SaaS development settings. The lean theory is based on the need to minimize waste, enhance flow and to concentrate on creating value as seen by the customer. The identified inefficiencies in the existing workflow are more or less closely related to the types of waste outlined in the literature on Lean, specifically waiting, overprocessing, and excessive work in progress (Womack and Jones, 2003).

The additions made to the state workflow of the future are echoed in the ideals of Kaizen, which focuses on gradual improvement and continuous improvement. Instead of re-architecturing the entire development process, the changes were based on minor ones, including the introduction of backlog triage meetings, a decreased batch size, and enhanced initial team-to-team collaboration. These transformations demonstrate that even minor adjustments to the processes could result in a great workflow optimization over time.

The other significant finding of the empirical results is the contribution of visualization in process improvement. Value Stream Mapping assisted the stakeholders to see the flow of tasks in the system and where delay can be found. The fact that the entire workflow was visualized helped the teams determine any inefficiencies that could not be seen at a daily operation. It is a proven fact that visualization tools are very effective tools to enhance organizational learning and decision making in the Lean management environment (Rother and Shook, 2003).

Kaizen philosophy, together with Value Stream Mapping, is thus a potent one-stop-shop of continuous improvement in knowledge-based organizations. These strategies played a role in Company X, as they assisted the stakeholders to shift their implicit knowledge of workflow issues to a formal analysis of performance of processes. This change allowed the organization to determine specific improvements that would have huge impact in terms of decreasing lead time and enhancing teamwork.

4.9 Chapter Summary

This chapter highlighted the empirical results of the research in terms of interviews with stakeholders of Company X and the examination of internal processes with the help of Value Stream Mapping. The evaluation showed that the current workflow process is effective to facilitate coordination among the Product, Engineering, QA, as well as Technical Support Engineering teams, but it has serious inefficiencies concerning the waiting time, work in batches, and slow communication.

The analysis of the Value Stream Mapping showed that the existing workflow was taking around 307 days to develop the feature and only 96 days were value-add development processes. Most of the overall lead time was comprised of waiting times between stages of the workflow. Using the Lean and Kaizen, a new state workflow was created that would greatly cut these delays and enhance team coordination.

The future state workflow shortened the overall lead time to about 84 days and boosted the process efficiency by 31 percent to 59 percent. These were incremental improvements including backlog triage processes, smaller work batches, continuous estimation, better coordination between Product and Engineering teams and more efficient testing processes and release processes.

The results of this chapter illustrate that Lean process analysis and continuous improvement practices can be effectively implemented in the SaaS development settings. These findings are discussed in the following chapter in more detail and their implications on project management, optimization of workflow, and the practices of continuous improvement in the software organization are discussed.

Chapter 5: Discussion and Conclusion

5.1 Introduction

This is the last chapter in this thesis, which discusses and gives the conclusions of the thesis. This chapter is aimed at interpreting the empirical results of the research and relating them to the theoretical framework outlined in the previous chapters. Although the last chapter was devoted to the presentation of the results of the Value Stream Mapping exercise and the observed workflow characteristics, the present chapter interprets the meaning of these results and how they relate to theory and practice. Specific attention is paid to the role of Lean thinking, the principles of Kaizen, and the transparency of the workflow as the factors that help to increase the efficiency of the processes within the SaaS product development setting.

The aim of the thesis was to assess the workflow of feature improvement request in Company X and to determine the possibility of enhancing the efficiency by applying the continuous improvement frameworks. The study of the interplay between Product, Engineering, Quality Assurance, and Technical Support Engineering teams was also designed to gain insight into the effect of cross-functional collaboration on the workflow performance and the ability to implement incremental refinements to minimize delays in the complex development processes. The findings show that using rather minor changes in workflow design and coordination practices, it is possible to make major gains in lead time and process efficiency.

5.2 Discussion of Key Findings

One more valuable lesson that the study has brought to mind is connected with the correlation between the visibility of the workflow and organizational learning. The Value Stream Mapping exercise was not only a measurement tool, but also a communication mechanism, which enabled stakeholders in various departments to see the entire workflow through to the end. In most organizations, the employees only know the part of the process that they are directly involved in.

This divided knowledge may cause one to find it hard to appreciate the impact of delays at one point to the whole system. The Value Stream Map gave stakeholders an opportunity to see how their work relates to other teams as it gave them a visual representation of the entire workflow. Such awareness of shared process is also frequently cited in Lean literature as a crucial element in continuous improvement since it allows teams to detect inefficiencies as a group instead of referring to their own roles and duties (Hines et al., 2004).

Another significant point based on the findings is that the flexibility and structure of the development of SaaS workflows should be balanced. Agile development methodologies have been put forward with the focus on flexibility and incremental delivery, whereas at times over flexibility may lead to a loose definition of processes where the responsibilities and priorities are not clearly defined (Boehm and Turner, 2004). The existing workflow monitoring in Company X revealed that despite the fact that teams were capable of adapting to the evolving customer needs, there were no rigid restrictions of the backlog size and work-in-progress rates, which resulted in huge queues of unfinished tasks awaiting completion.

The future state workflow resolved this problem by establishing lightweight coordination structures including backlog triage meetings and work-in-progress limits. These mechanisms are useful in keeping things flexible and yet giving sufficient structure so that the tasks pass through the workflow in a manner that is efficient.

The other important finding of the study relates to how the quality of communication can help to minimize the delays in the workflow. The stakeholder interviews revealed that not all the delays witnessed in the current workflow were due to incomplete or vague information in feature requests. In case the requirements were not well defined, it meant that more communication cycles had to be made between Product, Engineering and QA teams to ensure that the expected functionality was understood.

These explanation loops added to the time it took to finish tasks and added to rework loops. In order to increase workflow efficiency, the enhancement of the clarity of requirements and acceptance criteria is thus an opportunity. This observation is in line with the past studies that indicate that clearly articulated requirements are important in minimizing the uncertainty in development and enhancing the coordination among cross-functional teams (Schwaber and Sutherland, 2020).

Another value shown by the results is the significance of thinking at the system level in assessing process performance. In most companies, the performance measurements are more about the output of a particular group, like the amount of work done or the speed of a development process. Although these metrics are helpful, they can miss systemic inefficiencies that can exist between workflow stages. The Value Stream Mapping analysis carried out in this paper showed that most of the delays were experienced between teams, as opposed to in individual tasks. This finding supports the Lean management philosophy according to which enhancing the entire system can yield more performance improvement than enhancing one component of the system separately (Ohno, 1988).

5.3 Linking Empirical Findings to Lean and Kaizen Principles

The empirical results of this research are easily explained in terms of the Kaizen concepts and Lean philosophy. Among the key findings was the existence of long waiting time between stages of the workflow especially in backlog management and testing. The waiting time is regarded as one of the main types of waste that do not add value to the process according to the Lean theory (Dingsøyr et al., 2012). The huge number of backlog items and the latent prioritization observed in the present workflow are thus the typical examples of process inefficiency on the Lean standpoint.

On the same note, the inefficiency in Lean systems is also evident in the use of batch processing in estimation, development and testing phases. Batches of large size cause more waiting time and a decrease in continuity of a flow that is directly related to overall performance of the process (Reinertsen, 2009). New features added to the state workflow in the future, including the reduction of work-batches and the ongoing estimation, are in line with the Lean principles of the workflow focusing on flow efficiency and minimizing work-in-progress.

The results are also very favorable to the use of Kaizen as a continuous improvement approach. Instead of suggesting any radical changes to the working process, the innovations created in this study were aimed at the introduction of incremental changes, i.e., the betterment of the backlog triage, the strengthening of the initial cooperation between the teams, and the establishment of the constant testing routines. This strategy embodies the very essence of Kaizen, according to which minor and ongoing enhancements result in major long-term improvements in performance (Sommerville, 2016).

Moreover, the fact that the future state workflow is being developed jointly with stakeholders also reflects one more significant element of Kaizen, i.e., the engagement of the employees in the process of improvement. The organization stands a good chance of implementation because by engaging various departments in finding areas of inefficiencies and suggesting ways of remedies, the chances of having better working conditions are high and can be maintained in the long term.

Overall, the alignment between empirical findings and Lean and Kaizen principles indicates that the observed inefficiencies are not unique to a single organization but are consistent with broader process improvement theory. This strengthens the validity of the study and demonstrates that the proposed improvements are grounded in established theoretical frameworks.

5.4 Organizational Validation of the Proposed Workflow Improvements

The organizational response to the suggested workflow improvements that were created as the result of the Value Stream Mapping exercise was an important outcome of this research. The presentation and discussion of the future state workflow and suggested improvement mechanisms during the collaborative review with the stakeholders of Company X included Product, Engineering, Quality Assurance and Technical Support Engineering representatives. This discussion had the aim of assessing whether or not the proposed improvements were deemed to be viable in the operational context of the organization, and whether they were consistent with current development practices.

The stakeholders were found to be in agreement with the broad direction of the proposed improvements and acknowledged that the future state workflow was going to deal with many of the inefficiencies that were identified during the current state analysis. Specifically, the participants recognized that waiting too long in the backlog stage, work in large batches at the design phase, late estimation sessions, and bottlenecks in testing were some of the areas where process enhancements could yield significant improvements in performance.

Consequently, the organization stated that the improvement proposals would be discussed internally with the aim of progressively adopting some of the recommended practices in the current workflow framework.

To be more precise, the Company X took corroboration that the enhancement suggestions associated with four major workflow phases would be analyzed to be put into practice. These phases encompass backlog management, product and design collaboration, estimation practices and testing processes. To manage backlog, the suggested addition of a backlog triage meeting and work-in-progress limits was viewed as a potentially effective tool to minimize the number of outstanding feature requests and enhance the visibility of prioritization. During the product and design phase, the idea of reinforcing the initial interaction between Product and Engineering teams was considered a successful method of minimizing the requirements clarification processes and enhancing the technical viability of design decisions.

On the same note, the suggestion to implement more continuous estimation practices was recognized as the possible sphere of improvement in the current planning procedure. Rather than depending on the planned estimation meetings, the organization showed an interest in testing incremental estimation methods that would enable the feature requests to be estimated faster as they proceed through the development pipeline. Lastly, the suggestions offered towards the testing phase, such as a more active application of continuous testing activities and a higher degree of automatization of testing tools, were identified as solutions that would allow minimizing the bottlenecks in testing and enhancing the overall flow of work.

Though the introduction of such improvements will be a gradual process and will be determined by the internal organizational priorities, the fact that the suggested workflow changes were received positively proves the everyday applicability of the obtained research results. The fact that the stakeholders in the organization recognized the importance of the suggested improvements means that the Value Stream Mapping analysis was able to identify some important inefficiencies in the development workflow. Moreover, the fact that the organization is willing to assess such improvements indicates that the research can be used practically in areas beyond the academic field of the thesis.

Research wise, this outcome makes the research more valid since it will clearly show that the improvements proposed were not only theoretically based, but also realistically thought by practitioners who were directly involved in the workflow. The fact that organizational validation of improvement proposals was made is an important sign that the research findings are based on the real problems in operation and not just theoretical observations. Consequently, the research not only adds to the theoretical knowledge of workflow optimization in a SaaS but also to the real organizational learning in Company X.

5.5 Theoretical Implications

This research can be used to supplement the literature on Lean process improvement by illustrating how Value Stream Mapping can be applied to knowledge-intensive processes. Conventional Lean implementations target the manufacturing processes where the value is created by transforming tangible materials. Software development settings on the other hand add value by processing and exchanging information. The nature of the work is also quite different, but the principles of the Lean thinking are still applicable. Even in situations where the work is information-based and not a physical product, waiting time, too much work-in-progress, and ineffective handoffs between stages can still cause delays.

The results of this study thus contribute to the emerging literature that utilizes the Lean principles in software development settings. By modifying Value Stream Mapping to focus on the flow of information, instead of the flow of materials, organizations are able to recognize inefficiencies that would otherwise go unnoticed in the complex digital workflow. The case that is discussed in this paper shows how the process of visualizing the information flow between Product, Engineering, QA and Technical Support Engineering teams can help to identify coordination issues that cause long lead times.

The other theoretical implication is associated with the notion of continuous improvement as an organizational ability. Kaizen philosophy underlines that improvement cannot be a one time project but a continuous process that must be incorporated into everyday work activities. This philosophy is represented in the improvements made in the future state workflow, which revolves around minor adjustments that can be easily made but will increase the efficiency of the processes over time. The organization made specific changes to the workflow, including the use of backlog management, previous cross-team work, and smaller development cycles, instead of redesigning the entire process. All these incremental changes had a cumulative effect in achieving a significant decrease in the overall lead time.

The study also adds to the research on cross-functional cooperation in software development. Numerous studies highlight the need to work closely between product management and engineering teams, whereas little research focuses on the interplay between development teams and support-related functions. With the introduction of Technical Support Engineering in the study, the value of the support teams in authorizing customer requests and offering technical expertise to product problems is emphasized. With the addition of TSE to the workflow analysis, the study gives a more in-depth insight into the way customer feedback is incorporated into the process of product development.

5.6 Practical Implications

The implication of this research to practice does not just apply to the case organization as was studied in the research. A number of SaaS companies are established in the environment of continuous product development, frequent customer feedbacks and varying feature requirements

(Ries, 2011). In such an environment, having a streamlined workflow is very important in making sure that new features and improvements are delivered to the customers within a very short time. The findings of this study provide several practical lessons that organizations can adopt to improve their developmental processes.

Firstly, the research indicates the importance of the visual evaluation of the workflow performance periodically, which can be achieved with the assistance of such visualization tools as Value Stream Mapping. The organizations tend to informally perceive performance of the processes rather than analysing them systematically. Mapping the entire workflow and computing the time to process the work and the waiting time will assist organizations to know better where the inefficiencies are. This analysis would provide an evidence-based foundation to any improvement initiatives, and will help ensure that the observed evidence is the basis of the changes in the process, not assumptions.

Second, the research indicates how crucial the management of work-in-progress within development pipelines is. Backlog accumulation and huge lines of outstanding work can significantly contribute to lead time and reduce system responsiveness. Such mechanisms as backlog triage meetings, WIP limits can be applied to assist organisations in managing the manageable pipeline of tasks and requests that can be assessed quicker.

Third, the research reveals the importance of cross-functional cooperation in early age. Consulting technical stakeholders at a more initial stage of the requirement definition process allows potential technical constraints to be established at an earlier stage before the detailed design process itself begins. This helps to reduce the possibility of rework during the later stages of development and helps to ensure that design decisions do not contradict technical feasibility.

The other helpful implication is linked with the value of automation in streamlining the workflow. Teams can use the continuous integration pipelines and automated testing tools to detect the issues in the earlier development stage and reduce the amount of manual efforts required to conduct the verification processes. The process of releasing is also automated and hence more regular releases to the customers can be done with the completed features. These technologies are therefore useful in the support of Lean and Kaizen principles in the modern software development environments.

5.7 Organizational Learning and Continuous Improvement

The other observation that the study has yielded is in relation to the relationship between organizational learning and workflow analysis. The Value Stream Mapping exercise, which was conducted on the present study, not only assisted in the identification of the inefficiencies but also made the stakeholders think about the value their work contributed to the development process as a whole. When teams are able to view the entire workflow rather than focusing on their individual jobs, they are able to come up with a more holistic picture of how their activities influence the work of the entire system.

This is the shared ground that assists in creating a culture of constant enhancement in the organization. The stakeholders will begin to think about workflow issues as systemic problems that need to be resolved in a coordinated manner and not as individual issues in some teams. This school of thought coincides with the ideas of Kaizen that lay emphasis on shared responsibility of process enhancement and encourages the participation of all employees in process refinement through discovery of opportunities.

The collaborative nature of the workflow enhancement process that was witnessed in this research study means that the success of effective workflow enhancement initiatives must be pegged on not only analytic tools, but also on the organizational culture. They will be more likely to aid in the process of implementing the changes and making the process remain refined by actively involving the stakeholders in the analysis of the workflows and designing the improvements to the workflow.

5.8 Limitations of the Study

Despite the fact that the study offers great insights into how the workflow can be improved in SaaS organizations, there are a number of limitations that should be considered. The study is done with one case organization, which restricts the application of the results to other industries or organizational structures. The specific improvements that have been revealed during the course of the current study may not be relevant to all organizations because SaaS companies can differ greatly in terms of team structure, practices in development and the complexity of their products.

The other weakness is associated with the qualitative research methodology. Although interviews and workflow observations were excellent sources of information about organizational practice, the analysis is subjective in part as it is based on the interpretations of stakeholders and the researcher.

Despite the attempted triangulation of data by document analysis and Value Stream Mapping, there is a certain level of subjectivity in qualitative research.

The fact that the researcher is an employee and a researcher in the organization is another issue that may pose challenges concerning objectivity. Although this role was a great opportunity to have access to internal workflow and documentation, it was necessary to take a step back so that the analysis is unbiased and balanced. This risk was averted through transparency of the research design and the use of numerous data sources.

5.9 Recommendations for Future Research

The findings of this research can be expanded in the future by examining workflow optimization programs of different SaaS enterprises. Case study comparison can be an efficient chance to learn how other organizational structure and development practices impact the efficiency of workflow. Such studies can be effective in terms of this research because perhaps it can be concluded that the findings of this study are generalized or the findings are limited to peculiarities of organizations.

The second potential future research path is to add quantitative performance measurements to the qualitative workflow analysis. A good starting point in process flow is Value Stream Mapping, which can be supplemented with more focused development metrics to provide a more detailed view of how improvements in the workflow would impact the overall performance of the software development process in terms of cycle time, defect rates, and deployment frequency.

The application of the new technologies in the support of Lean and Kaizen practices in software development environments might also be a research topic in the future. As the practices of DevOps constantly evolve, automated deployment pipelines, continuous monitoring, and AI-based development tools might also be getting more useful in facilitating the workflows. One of the areas of interest that can be studied is how these technologies will interact with the organizational processes.

That the improvements to the workflow presented by the authors were well-received by the stakeholders in Company X is another sign that the findings of the research are providing valuable information that can be used to facilitate the real changes in the organizational processes.

References (APA 7)

1. Womack, J. P., & Jones, D. T. (2003). Lean thinking: Banish waste and create wealth in your corporation. Free Press.
2. Ohno, T. (1988). Toyota production system: Beyond large-scale production. Productivity Press.
3. Rother, M., & Shook, J. (2003). Learning to see: Value stream mapping to add value and eliminate muda. Lean Enterprise Institute.
4. Liker, J. K. (2004). The Toyota way: 14 management principles from the world's greatest manufacturer. McGraw-Hill.
5. Poppendieck, M., & Poppendieck, T. (2003). Lean software development: An agile toolkit. Addison-Wesley.
6. Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., & Thomas, D. (2001). Manifesto for agile software development. https://agilemanifesto.org
7. Hines, P., Holweg, M., & Rich, N. (2004). Learning to evolve: A review of contemporary lean thinking. International Journal of Operations & Production Management, 24(10), 994–1011.
8. Sommerville, I. (2016). Software engineering (10th ed.). Pearson.
9. Kerzner, H. (2017). Project management: A systems approach to planning, scheduling, and controlling (12th ed.). Wiley.
10. Anderson, D. J. (2010). Kanban: Successful evolutionary change for your technology business. Blue Hole Press.
11. Reinertsen, D. (2009). The principles of product development flow: Second generation lean product development. Celeritas Publishing.
12. Schwaber, K., & Sutherland, J. (2020). The Scrum guide. Scrum.org. https://scrumguides.org/scrum-guide.html
13. Humble, J., & Farley, D. (2011). Continuous delivery: Reliable software releases through build, test, and deployment automation. Addison-Wesley.
14. Kim, G., Humble, J., Debois, P., & Willis, J. (2016). The DevOps handbook. IT Revolution Press.
15. Forsgren, N., Humble, J., & Kim, G. (2018). Accelerate: The science of lean software and DevOps. IT Revolution Press.
16. Fitzgerald, B., & Stol, K. J. (2017). Continuous software engineering: A roadmap and agenda. Journal of Systems and Software, 123, 176–189.
17. Edmondson, A. (2018). The fearless organization: Creating psychological safety in the workplace for learning, innovation, and growth. Wiley.
18. Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. (2012). A decade of agile methodologies: Towards explaining agile software development. Journal of Systems and Software, 85(6), 1213–1221.
19. Denning, S. (2018). The age of agile: How smart companies are transforming the way work gets done. AMACOM.
20. Flyvbjerg, B. (2014). What you should know about megaprojects and why: An overview. Project Management Journal, 45(2), 6–19.
21. Imai, M. (1986). Kaizen: The key to Japan’s competitive success. McGraw-Hill.
22. Highsmith, J. (2002). Agile software development ecosystems. Addison-Wesley.
23. Cohn, M. (2004). User stories applied: For agile software development. Addison-Wesley.
24. Yin, R. K. (2018). Case study research and applications: Design and methods (6th ed.). SAGE.

25. Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. Qualitative Research in Psychology, 3(2), 77–101.

26. Sutherland, J. (2014). Scrum: The art of doing twice the work in half the time. Crown Business.

27. Mann, D. (2005). Creating a lean culture: Tools to sustain lean conversions. Productivity Press.

28. Ries, E. (2011). The lean startup: How today’s entrepreneurs use continuous innovation to create radically successful businesses. Crown Business.

29. Boehm, B., & Turner, R. (2004). Balancing agility and discipline: A guide for the perplexed. Addison-Wesley.

30. De Luca, J., & Coad, P. (1999). Java modeling in color with UML: Enterprise components and process. Prentice Hall.

APPENDICES

Appendix A

Stakeholders Interview Analysis

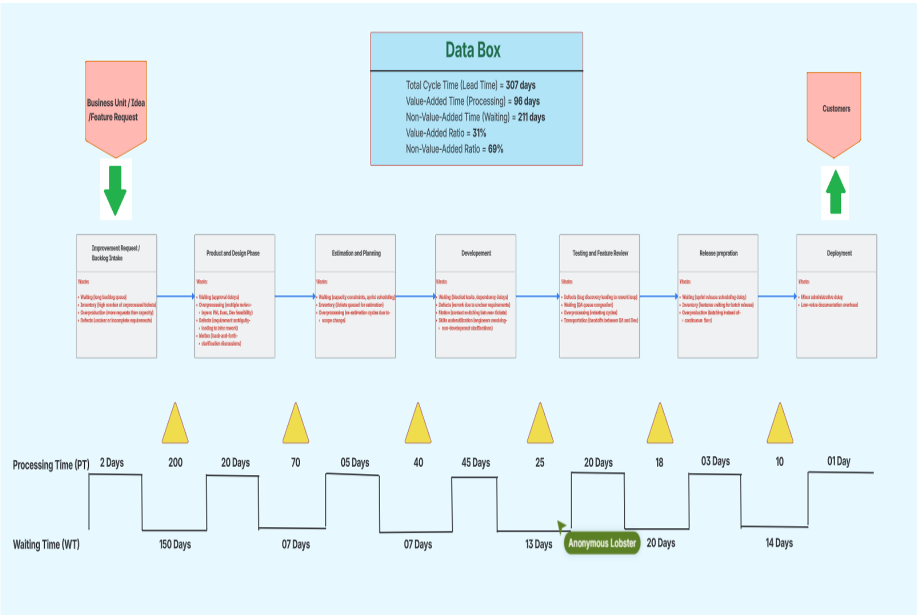
Table A1: Stakeholders Interview Analysis

Stakeholder	Role / Department	Analytical Perspective	Key Insights	Identified Issues	Contribution to Improvements
Stakeholder 1	Technical Product Manager	Thematic analysis (requirements & prioritization)	Emphasized importance of clear requirements and customer value alignment; backlog contains many unrefined items	Large backlog size, unclear requirements, delayed prioritization	Suggested backlog triage meetings, improved requirement clarity, early collaboration with Engineering
Stakeholder 2	VP of Engineering	Process and workflow analysis (development & testing stages)	Highlighted delays due to unclear specifications and batch testing; context switching affects efficiency	Rework due to unclear requirements, testing bottlenecks, batch processing delays	Recommended smaller work batches, continuous testing, early technical involvement in design stage
Stakeholder 3	Manager, Technical Support Engineering	Validation and intake analysis (issue identification & filtering)	Focused on importance of validating requests before Product stage; many requests lack sufficient detail	Invalid or unclear tickets, repeated clarification cycles with Product	Proposed stronger ticket validation, better communication with Support, improved initial data quality

Appendix B

AS IS Value Stream Mapping

Figure B1: AS IS Value Stream Mapping



Appendix B

Future Value Stream Mapping

Figure B2: Future Value Stream Mapping